ENGR7019 Engineering Dissertation Project Final Report

Simulation of Autonomous Vehicle in a Virtual Environment

Author (Rajkumar Iyer, 19189334)

Programme: MSc Automotive Engineering with Electric Vehicles

Module: MSc Dissertation

Academic year: 2022-23

Word Count	Main Part	Appendix
	7659	1557
Number of illustrations -	Main Part	Appendix
	30	7

School of Engineering, Computing and Mathematics

Statement of Originality

Except for those parts in which it is explicitly stated to the contrary, this project is my own work. It has not been submitted for any degree at this or any other academic or professional institution.

19/09/2023

--- ---

Signature of Author Date

Regulations Governing the Deposit and Use of Master of Science Dissertations in the School of Engineering, computing and Mathematics,

Oxford Brookes University.

1. The 'top' copies of projects submitted in fulfilment of Master of Science course requirements shall normally be kept by the Department.
2. The author shall sign a declaration agreeing that, at the supervisor's discretion, the dissertation may be submitted in electronic form to any plagiarism checking service or tool.
3. The author shall sign a declaration agreeing that the dissertation be available for reading and copying in any form at the discretion of either the project supervisor or in their absence the Head of Postgraduate Programmes, in accordance with 5 below.
4. The project supervisor shall safeguard the interests of the author by requiring persons who consult the dissertation to sign a declaration acknowledging the author's copyright.
5. Permission for anyone other than the author to reproduce in any form or photocopy any part of the dissertation must be obtained from the project supervisor, or in their absence the Head of Postgraduate Programmes, who will give his/her permission for such reproduction only to the extent which he/she considers to be fair and reasonable.

I agree that this dissertation may be submitted in electronic form to any plagiarism checking service or tool at the discretion of my project supervisor in accordance with regulation 2 above.

I agree that this dissertation may be available for reading and photocopying at the discretion of my project supervisor or the Head of Postgraduate Programmes in accordance with regulation 5 above.

19/09/2023

--- ---

Signature of Author Date

Abstract

Scenario-based testing using simulators has become a standard practice in the automotive industry after being established as a reliable and cost-effective model and concept validation technique. The research community has also acknowledged the significance of this approach, considering the variety of advantages it has to offer. However, a discussion on the simulation model's accuracy and uncertainties is inadequately highlighted in the context of using simulators to test autonomous vehicle technology. Any conclusions regarding an autonomous system's functionality are incorrect without considering the model's limitations and accuracy.

The project “Simulation of Autonomous Vehicle in Virtual Environment” presents a comprehensive study focused on high-fidelity modelling and validation of an autonomous vehicle to measure the virtual environment's effectiveness, realism, and reliability for testing and development.

The study outlines a detailed approach to developing and integrating high-fidelity models while considering real-world considerations to build and replicate real-world scenarios convincingly within the simulation to ensure that the vehicle developed in the virtual environment experiences the same challenges and constraints as it would in an actual scenario.

Validation efforts demonstrate a high degree of correlation between simulation outputs and actual sensor data, reinforcing the reliability of the simulated environment. The enhanced accuracy of sensor models and validated simulation results reduces the dependency on physical testing, allowing for accelerated testing and deployment of autonomous vehicles. The assessment of simulation software effectiveness reveals insights into its limitations and strengths, aiding in software selection for specific needs.

The project contributes to the study of developing a robust simulation framework, including high-fidelity modelling, simulation and validation. Based on a thorough analysis of the simulation results, a detailed discussion on evaluating the effectiveness of the simulation software is presented, along with suggestions for prospects. As the automotive industry continues to evolve, the findings from this project offer valuable insights for researchers, engineers and developers working on the simulation of autonomous vehicles.

Highlights

- Development of an advanced simulator, including realistic and accurate vehicle, sensor and environment model and integrated with the Autonomous Driving System developed by OBRA using ROS2.
- Comprehensive validation of the developed vehicle and sensor model.
- Performed Software-In-the-Loop, Hardware-In-the-Loop, and Vehicle-In-the-Loop simulation.
- The project's concept and approach were presented during the FS-AI UK Simulation Development presentation and won the 1st Rank.

Table of Contents of Main Part

Table of Contents of appendix

List of Figures of main part

List of Tables of main part

List of Figures in Appendix

List of Tables of Appendix

Glossary

- **Autonomous Vehicle (AV)**
- **Robot Operating System (ROS)**
- **Validation**
- **Simulation**
- **Virtual environment**
- **Sensors**
- **Scenario**
- **Path Planning**
- **Simultaneous localisation and mapping (SLAM)**
- **Autonomous Driving System (ADS) –** *The ADS in this project refers to the software stack that enables the autonomous functioning of the vehicle*
- **Autonomous Driving System - Dedicated Vehicle (ADS-DV) –** *The ADS-DV is the test vehicle designed and developed by the FS competition organiser to test and validate the ADS developed by the participating teams.*
- **Autonomous Driving System - Buggy Vehicle (ADS-BV)** – *The ADS-BV test vehicle is a close replica of the ADS-DV, designed and developed by OBR A to tackle the primary vehicle's availability issue and aid in developing the ADS system.*

1. Introduction

1.1 Background

In the era of rapid technological advancements, the automotive industry stands on the brink of a transformational change with the introduction of autonomous vehicles (AVs). The concept of a self-driving vehicle has emerged as a revolutionary idea and has gained significant interest not just within the industry but also among government, academia, and the general public, weighing up its potential to transform mobility (Teng, et al., 2021).

Although autonomous vehicle technology has been studied and developed extensively, safety assessment and type approval have caused a delay in their deployment. To ensure the safe operation of the AV, it needs to be evaluated under extensive scenarios, which is an expensive and time-consuming process (Shah, et al., 2017). Moreover, the conventional testing method also raises safety concerns due to the unpredictable and unvalidated behaviour of the vehicle.

To overcome these limitations, scenario-based simulator testing has emerged as an ideal solution (Riedmaier, et al., 2021). This strategy enables the review of critical components of the AV, such as object detection, object classification, decision-making algorithms and vehicle behaviour within a controlled and scalable environment (Esteller, 2020).

The results obtained from the simulation provide valuable insights into the performance metrics, enabling the developer to fine-tune and optimise their system by identifying potential faults and inefficiencies at an early stage of development.

However, to test and validate the performance of the Autonomous system in the virtual platform, it is essential to develop a test environment with a greater resemblance to the actual scenario considering real-world factors. Besides, it needs to be accompanied by model validation to achieve credibility.

1.1.1 Project Aim and Objectives

The project focuses on developing a comprehensive simulation framework that mimics real-world scenarios for evaluating and validating the performance of an autonomous vehicle in the virtual environment. The project aims to study and demonstrate the effectiveness of using a simulator to advance the development and testing of autonomous vehicle systems in a more efficient, reliable and controlled environment.

Objectives

OBJECTIVE	WHAT IS MEASURED?	HOW IS IT MEASURED?	WHAT IS THE TARGET?
SELECTION OF AN IDEAL SIMULATION TOOL	Flexibility to develop a high-fidelity simulation model and replicate real-world scenarios.	Assessing the simulation software's capabilities to develop scenarios and incorporate environmental variables, vehicle models, sensor integration, etc...	Achieve a close resemblance of simulation with real-world test scenarios.
DEVELOP ACCURATE SIMULATION MODELS.	The accuracy and fidelity of the models.	Comparing simulation output with real-world data under controlled scenarios.	Achieve a high-level correlation coefficient.
ASSESS THE EFFECTIVENESS OF SIMULATION SOFTWARE.	How closely does the simulation resemble real-world scenarios and sensor interactions?	A comparative analysis is carried out by standardising situations in virtual and actual settings, and variables, including sensor outputs and vehicle and environmental dynamics, are examined.	Demonstrate a high level of consistency and similarity.
	How valuable the simulation software is for testing and development.	Performing SIL, HIL and VIL simulations.	To effectively test and validate both the concept model and hardware.
DISCUSSION AND RECOMMENDATIONS FOR ENHANCEMENT OF VIRTUAL VALIDATION	Insights and practical limitations associated with modelling and simulation.	Literature review, observation from the project and discussions from the journals.	To study the effectiveness of simulation software and areas of improvement.

Table 1 Objectives

1.2 Summary of Literature Review

According to the recent automotive trend, the demand and popularity for AVs have increased due to their potential to revolutionise the mode of transportation in terms of overall effectiveness, improved road safety, convenience, accessibility, comfort, etc. Moreover, considering the rapid increase in the number of on-road vehicles concerning the safety of people due to the increased likelihood of road accidents, the national administrations are likewise motivated to enhance the level of vehicle automation as a measure to prevent casualties (Riedmaier, et al., 2021). Since then, the advancement of autonomous vehicle technology has been among the most promising areas of research and development in the automotive industry.

Developing an AV is a multidisciplinary activity and must be verified effectively before deployment. However, the major challenge associated with verifying the system is the difficulty in testing due to multiple factors, including its un-predictive and random behaviour, dependability, technical complexity (such as the integration of sensors, perception, and algorithms), regulatory and legal frameworks, infrastructure needs, cost, affordability etc. As a result, it has become critical and essential to identify a more efficient and effective approach for validating the developed autonomous system. Virtual validation, a widely accepted practice used for estimating and validating the performance of a system/concept within a virtual 3D environment, emerges as an ideal alternative or a complementary method to the existing assessment technique for testing and validating the autonomous system (Choi, et al., 2021).

However, the scenarios used for testing the vehicle dynamics and powertrain areas are inadequate for testing autonomous vehicles. Requirements such as synchronised traffic patterns, traffic signals, traffic flow control, delays, etc., are significant hurdles in simulating an AV. Also, the other major challenge associated with testing the sensor models is the creation of high-fidelity environment objects, resulting in ineffective testing of the autonomous system.

In an AV, the most fundamental and critical component is the control system, which contains four main areas, i.e., perception, localisation, cognition, and motion control (Siegwart, et al., 2011). The Autonomous Driving System follows a systematic sense, plan, and act approach. The system must sense and understand the surrounding environment and determine its current state and position by the information obtained from the onboard sensor and plan accordingly aimed at the following action for the vehicle to reach the target and execute the plan (Schlager, et al., 2020). It is, therefore, necessary to train the autonomous vehicle to behave like a human, which requires training for countless scenarios and involves highly complex simulations. Therefore, developing an accurate simulation environment is crucial for simulating the autonomous system. The simulator platform required to generate a simulation environment for training and testing of autonomous vehicles needs to be flexible enough to develop high fidelity model to mimic real-world scenarios, which allows considering all the influencing parameters, including vehicle model, sensor modelling, environmental conditions, traffic, communication with an external platform which runs the control algorithm etc. (Shao, et al., 2022). As a result, it is vital to choose a simulator that allows the user to consider and work on the topics mentioned above.

Advantages of Using a Simulator

- Modelling and Validation of the concept
- Algorithm Development
- Training the autonomous system
- System Integration and verification
- Unit/component testing in HIL Simulation
- Regulatory compliance and certification

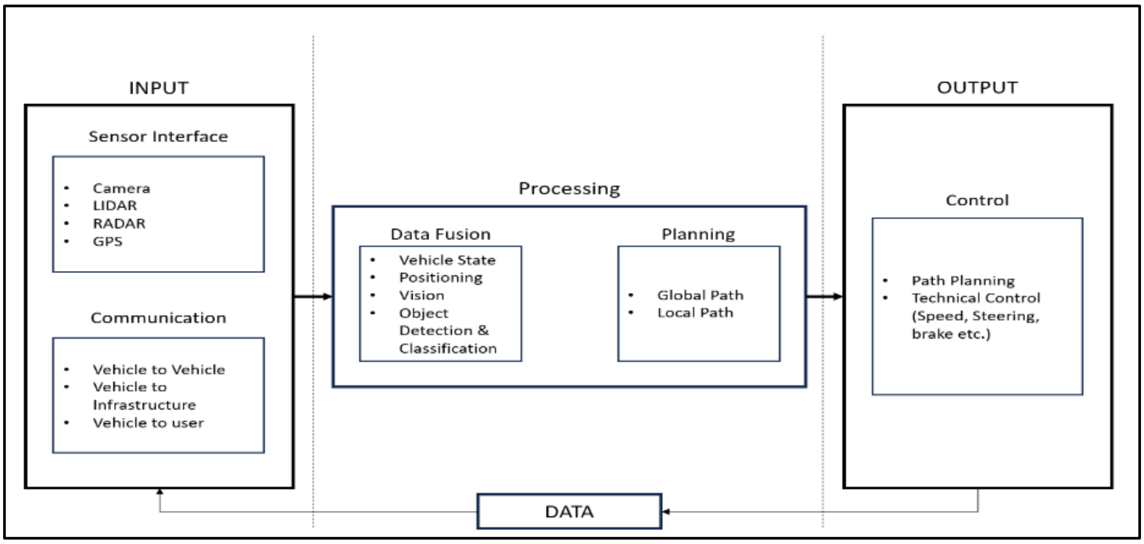


Figure 1 Autonomous Vehicle Architecture after (Bezai, et al., 2021)

There are existing State-of-the-art simulation software being used to test and validate a developed Autonomous Driving System. However, a considerable difference is still observed between the results of the real world and the simulated scenarios. As discussed in the research paper, the following factors might explain this variance.

1) Accuracy of the developed models: - The simulators do not scale well compared to actual tests due to limited consideration of parameters for modelling (Kim, et al., 2017).
2) Sensor Limitation and Noise: - The failure to consider sensor effects and sensor inaccuracy, such as sensor noise and occlusion (Schlager, et al., 2020).
3) Assumption and Simplification in generating scenarios: - Limited exposure to surrounding objects in simulation can lead to reduced complexity and uncertainties compared to the actual one, resulting in simplified scenarios (Muktadir, et al., 2022).
4) Calibration and Validation of sensor model.
5) Altering the algorithm for simulating the vehicle in the virtual environment.
6) Reproducibility of the environmental aspects in the virtual environment (Kim, et al., 2017).

and more.

Besides, the journal article (Aparow, et al., 2019) highlights and discusses the prospect of using simulators as part of the approval framework for certification of the AV/ADS before their deployment, in addition to not just restricting their use to testing and validating concepts.

Studies have been done in the past relating to the simulation of AVs, focusing on the topics mentioned above. However, little work has been reported (Cime, et al., 2021) on an integrated approach of modelling, simulating, and validating considering these points. To address this gap, in the following project, a study has been conducted that focuses on the different aspects of high-fidelity modelling for simulating AVs in a virtual platform to perform virtual validation, which could benefit and enhance the overall development process of the autonomous system.

1.3 Originality and Contribution

The critical aspect of this project involves the development of a sophisticated virtual environment alongside the high-fidelity vehicle and sensor model that emulates real-world scenarios for autonomous vehicles. Except for the control algorithm developed by the OBR-A team, all the other content presented in this project is an original contribution unless otherwise stated. The validation of the simulation models and subsequent discussion are based on real-world test outcomes obtained from experiments conducted by members of the OBRA team on campus and during the event.

The Oxford Brookes Racing Autonomous team will have access to this developed simulator to aid in their continued development.

1.4 The Approach

A systematic and commonly practised industrial approach is being followed to address and conduct a detailed study on the use and effectiveness of simulators. Considering the requirement of a vehicle model with a subsequent autonomous driving system, the formula student team autonomous vehicle is used as the reference concept for the following project.

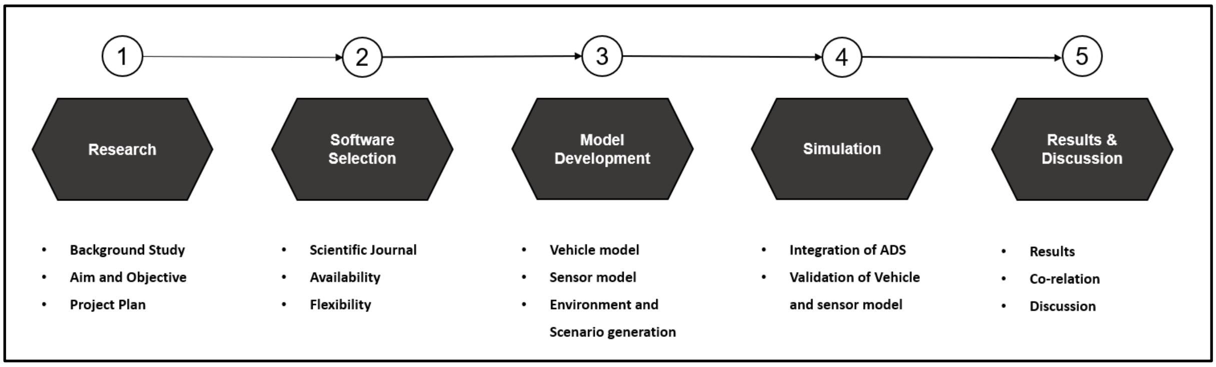


Figure 2 Approach Plan

2. Methodology

Simulating an autonomous vehicle within a virtual environment presents a difficult challenge due to its complex nature and the intricacy of various factors involved. To achieve the objectives of this project, a comprehensive and methodical approach was employed, as described in this section.

To extensively address the implemented methodology, it is organised into three distinct sub-sections

1. Software Selection
2. Modelling & Simulation
3. Validation & Discussion

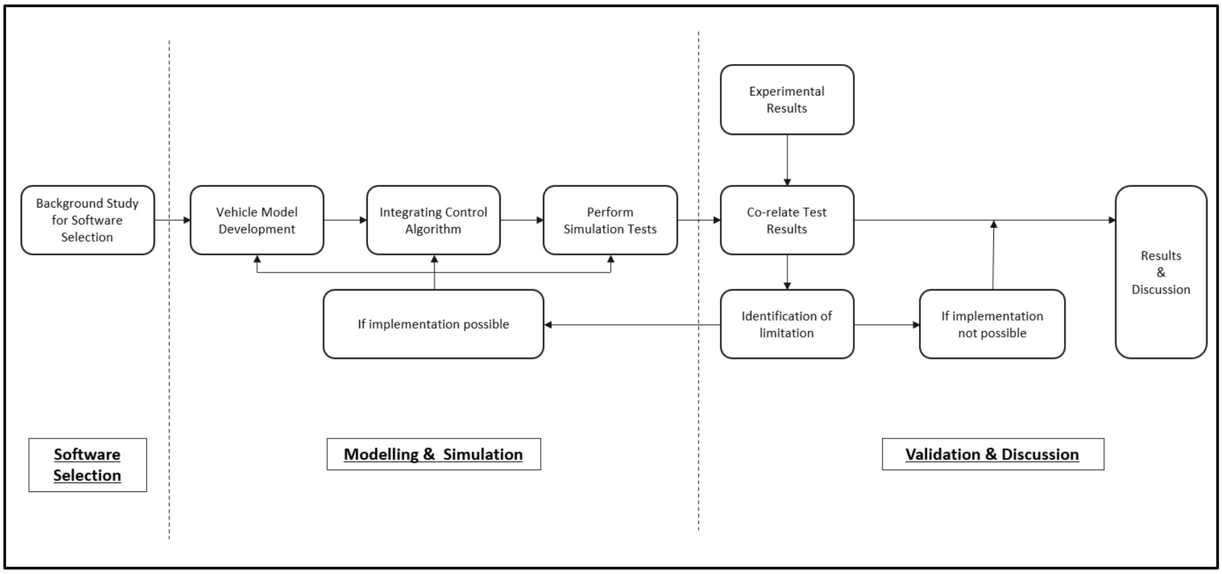


Figure 3 Methodology

2.1 Software Selection

The selection of a suitable simulation software was the fundamental step of this project. It demanded a comprehensive approach, as it directly influenced the simulation process's accuracy, competency and flexibility. To effectively address the criteria for selecting the software, it was necessary to comprehend the terms and reasoning of an autonomous vehicle's operation.

Perception, sensor fusion and path planning are the underlying aspects of the navigation of an autonomous vehicle. According to (Wu, et al., 2020), simultaneous positioning and map generation for autonomous navigation are made possible by combining localisation and mapping (SLAM) and path planning algorithms. This integration allows the vehicle to localise itself within the environment using the onboard sensors to create a map and plan a safe and efficient path to reach its target location.

Therefore, to simulate an AV in the virtual environment, it was essential to select a simulator that facilitates the development of a high-fidelity test environment with a dynamic vehicle and real-time sensor model that can deliver signals having relevant physical effects and offers the flexibility to be integrated with a platform which combines mapping and path planning algorithms to perform autonomous driving.

Integrating the autonomous vehicle control algorithm and simulation models demanded a middleware facilitating communication between components. ROS provided a robust framework for this purpose.

Robot Operating System (ROS)

ROS is an open-source middleware commonly used to develop robotic systems and control algorithms. ROS makes creating a modular, reusable component possible, allowing seamless incorporation of perception, localisation, planning and control modules into the simulation. In our project, ROS serves as the middleware that manages sensor data, perception algorithms and control commands for the AV. With this framework, the ADS uses sensor data from simulation software as inputs and produces control output based on the developed control algorithm/logic for simulating the vehicle autonomously in the virtual environment. For this project, ROS 2 foxy is being used.

It is important to note throughout this document that ROS2 will be referred to as ROS, not to be confused with its predecessor of the same name.

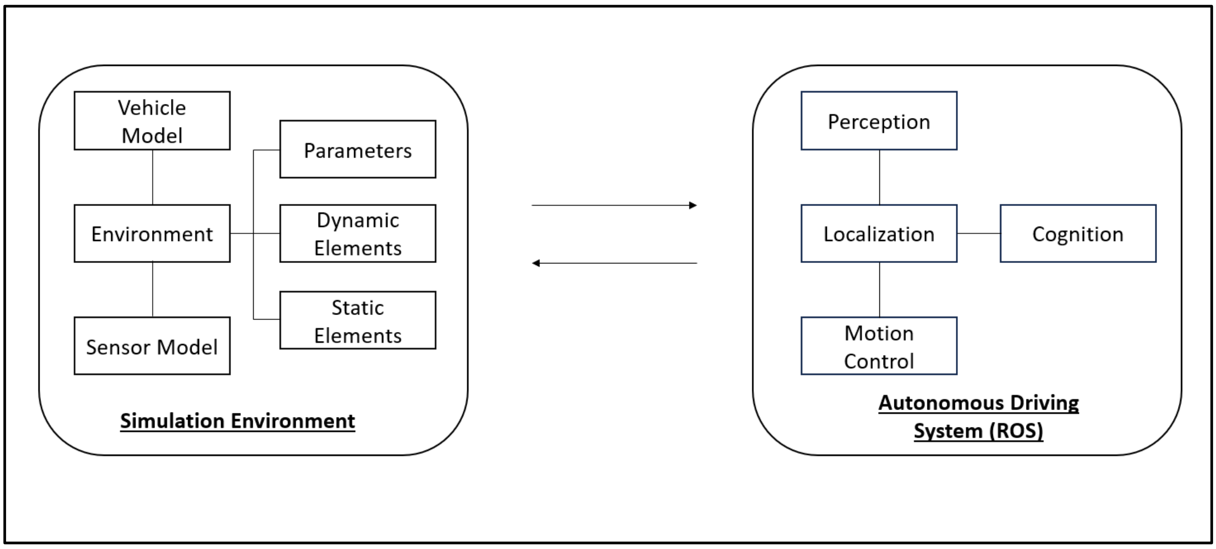


Figure 4 Simplified Architecture of the system for simulation of Autonomous Vehicle in Virtual Environment

Several specialised State-of-the-Art simulation software solutions are available on the market, which vary in model development, environment scaling, graphic quality, and feature sets (Platt & Ricks, 2022). However, based on the existing literature, as in (Rosique, et al., 2019), a comparative analysis was conducted on shortlisted simulation software options for achieving the project's objective.

CRITERIA/SIMULATOR	GAZEBO	UNITY	MATLAB/ SIMULINK	IPG CARMAKER	CARLA
ROS INTEGRATION	Yes	Yes	Yes	Yes	Yes
VEHICLE MODELLING	Yes - Basic	Yes	Y - High Fidelity	Y - High Fidelity	Yes
SCENARIO DEVELOPMENT	Yes	Yes	No	Yes	Yes
WEATHER	Yes - Post Processed	Yes	No	Yes - Post Processed	Yes
BRIDGE COMMUNICATION	Yes	Yes	Yes	Yes	Yes
REALISTIC/ VISUALIZATION	Medium	Yes	No	Yes	Yes
STABILITY OF THE SOFTWARE	Medium	Yes	Yes	Yes	Yes
SENSOR MODELLING	Yes	Yes - Limited	Yes	Yes	Yes
TRAFFIC OBJECT MODELLING	Yes - Difficult	Yes	No	Yes	Yes

Table 2 Comparison of Simulators

Given the above comparison, IPG Carmaker, a dedicated vehicle simulation software, had been considered for this study. The software's advanced physics simulation and integration capabilities offered robust sensor emulation and ROS compatibility, aligning with project goals for accurate and flexible vehicle simulation.

2.2 Module Design, Development and Simulation

When developing a high-fidelity test environment for simulation, it was essential to consider all the influencing factors that reflect real-world scenarios. These parameters were crucial in achieving high realism, accuracy and effectiveness in simulation. The key parameters focused on this project include:

Vehicle model, sensor model, geography and terrain, road infrastructure, environmental conditions and surrounding elements.

Vehicle Model Development

As the selected simulation software enabled the flexibility to develop and use a standard vehicle model for multiple test scenarios, the first step was to create a vehicle model in the simulation software, which included accurately represented physical components of the vehicle like chassis, tyre model, powertrain model, steering model, suspension model, C.G., etc. The accuracy of the model depended upon the availability of data. The parameterisation values required for modelling the vehicle had been considered from the datasheets provided by the event organiser who designed the car, and the additional parameters were measured in the lab from the vehicle in question.

Input Parameter

VEHICLE MODEL	
WHEELBASE	1.53 m
WHEEL TRACK	1.17 m
VEHICLE OVERALL HEIGHT	0.664 m
VEHICLE OVERALL LENGTH	2.8146 m
VEHICLE OVERALL WIDTH	1.43 m
VEHICLE MASS	306.5 Kg
VEHICLE REAR OVERHANG DISTANCE	0.563 m
DRIVE SYSTEM	
DRIVE MOTORS	Two Electric Motors
PEAK MOTOR TORQUE	55.7 Nm
PEAK MOTOR POWER	17 kW
MAX. MOTOR SPEED	4000 rpm
BELT DRIVE RATIO	3.5:1
PEAK MOTOR CURRENT	550 A
OPERATING VOLTAGE RANGE	31-90 V
TRACTION BATTERY	
MODEL	Chen
CELL CAPACITY	100 Ah
NOMINAL PACK VOLTAGE	51.2 V
OPERATING VOLTAGE	48 V
PEAK DISCHARGE CURRENT	800 A (up to 10 secs)

Table 3 Input parameters used for vehicle modelling.

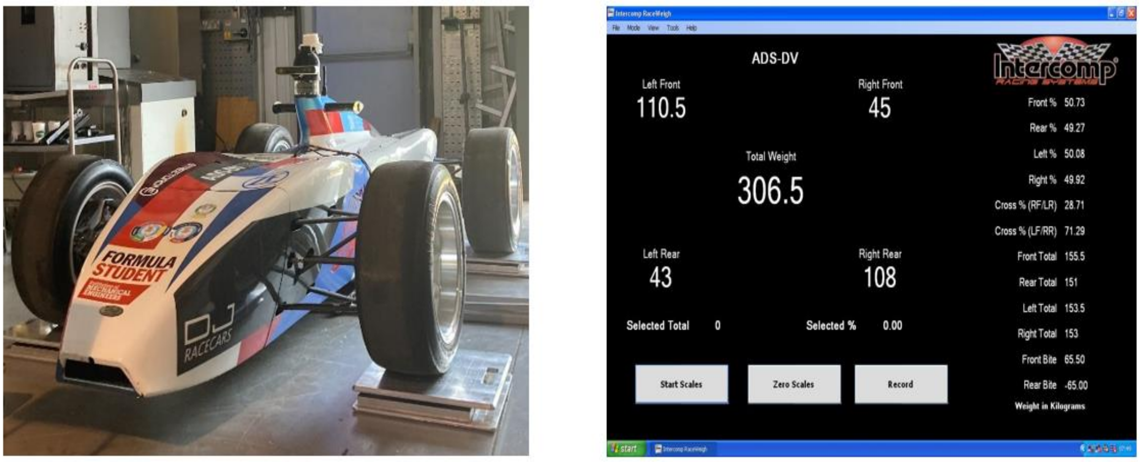


Figure 5 Physical measurement of ADS- DV

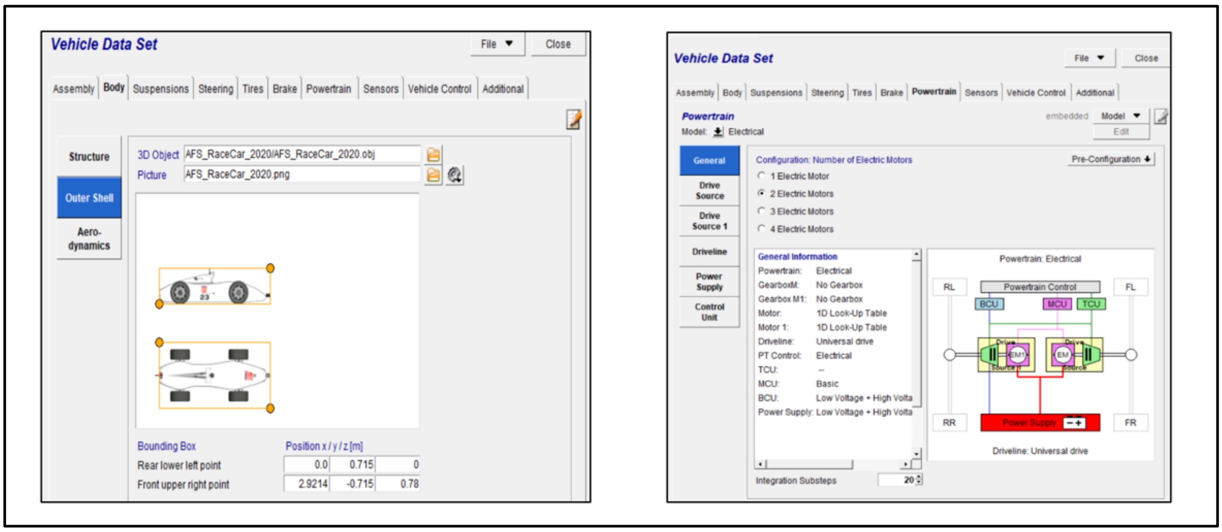


Figure 6 Vehicle Modelling in IPG Carmaker

Sensor Modelling

The autonomous car relies on various sensors onboard to realise its status and perceive its surroundings (Schlager, et al., 2020). These sensors work together and play a vital role in navigating the vehicle around the test environment and are categorised as Perception Sensor and Vehicle State Sensor.

a) Perception Sensor

These sensors serve as AV's primary information source, enabling them to see and sense their surroundings (Fursa, et al., 2021). They are responsible for providing a detailed understanding of essential environmental data for safe and reliable navigation. Since each sensor category has its own set of strengths and weaknesses based on its type and operating principle, depending on the application and requirement, a combination of a variety of sensors is usually used to overcome the limits associated with a single sensor type to produce an extensive picture of the environment (Schlager, et al., 2020). This fusion enhances the accuracy, reliability and redundancy, allowing the ADS to cross-verify information and make split-second decisions.

The reference model selected for this project uses only a Stereo camera as its perception sensor, and the same has been modelled. The IPG Carmaker software offers three different classes of sensor models of varying levels of detail, which help test various aspects of the system. This includes the Ideal sensor, HiFi sensor and Raw Signal Interface (RSI) sensor model (IPG Automotive , n.d.).

Sensor noise, resolution, mounting positions, orientation, FOV, occlusions, and other real-world occurrences were considered to ensure the simulation's accuracy.

b) Vehicle State Sensor

Alongside perceiving the environment, to perform autonomous navigation, the ADS also requires information about the vehicle's current state, including its position, orientation, movement and overall dynamic condition (steering wheel angle, wheel speed, brake actuator position, etc.). This information was gathered using sensors referred to as vehicle state sensors. A GPS sensor and the following in-built sensors are modelled to collect the above details.

- Inertial Sensor
- Slip Angle sensor

The built-in sensor model did not require parameterising but needed to be placed at appropriate locations. Information regarding the vehicle speed was directly available from the software.

CAMERA	ZED STEREO
OUTPUT RESOLUTION	608 * 352
FIELD OF VIEW	90(H)*60(V)*100(D)
FOCAL LENGTH	2.8 mm
BASELINE DEPTH	0.12 m
DEPTH	20 m
RANGE (RGB)	20 m
RANGE (DEPTH)	12 m

Table 4 Input Parameters Used for Sensor Modelling

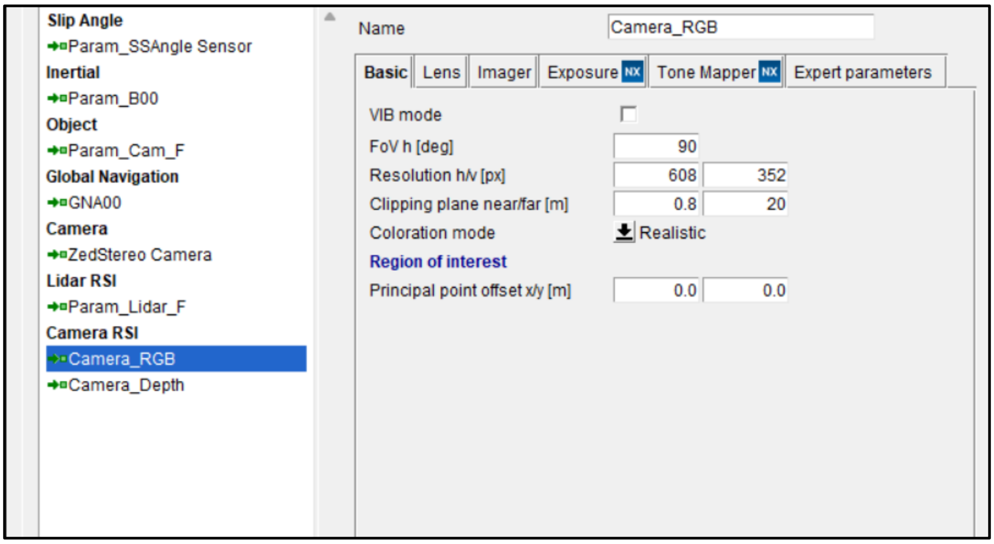
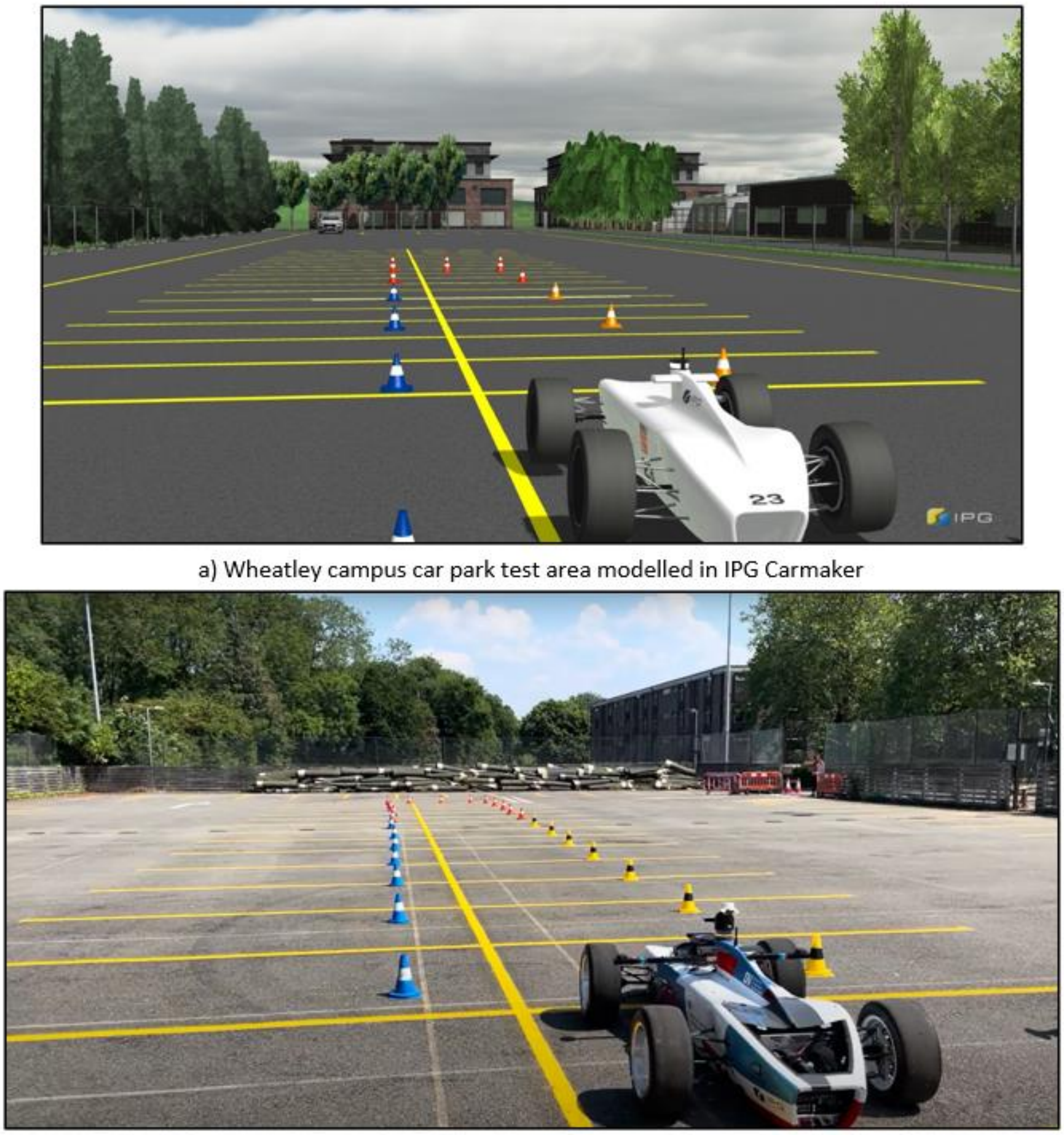


Figure 7 Sensor Modelling

Environment Model and Scenario Generation

The following step after developing the vehicle and sensor model was to create a test environment that closely resembles the actual test scenario. Since a formula student vehicle concept is being considered for the project where the navigation of the vehicle and point of interest only depends on the coloured cones around the track, the aim was to develop a test track with a detectable set of specific cones, as mentioned in the FS-AI rulebook. As a part of the competition tasks, the vehicle was subjected to compete in the acceleration, skid-pad, and autocross event; therefore, the following tracks were developed in the simulation environment using the CarMaker scenario editor. IPG Movie offered the option to change the dynamic positioning of the sun to alter overall lighting effects, which influenced the sensor model's perception abilities, enabling to perform tests under extensive scenarios. The simulation software also permitted the development of a test track using GPS coordinates, as in ***Figure 25 A4***.



a) Wheatley campus car park test area modelled in IPG Carmaker

b) Wheatley campus car park test area in Real-world

Figure 8 Environment Modelling (Wheatley campus test track recreated in a simulation environment)

Implementation of Controls

The integration of ROS and IPG Carmaker formed the foundation of our simulation framework. This enabled the synchronisation of sensor data, control algorithms and simulation environment to allow realistic and dynamic autonomous vehicle testing. This was achieved by developing ROS nodes, which represented the individual component modules of the system (Kato, et al., 2015). These nodes used topics (which manage input and output data) to communicate and interact with each other, following the popular publish-subscribe design pattern (Quigley, et al., 2015) to orchestrate the data flow in the system.

When the vehicle is placed in an active simulation environment, the sensor models in the simulator generate sensor data, i.e. 2D camera images, and publish this data as a ROS message on appropriate topics. Conversely, the node that had subscribed to these topics receives these messages, subsequently processes the data and estimates the 3D position of the cones. Using this information, the path planning and controls node plans a path and publishes a control command as a message on a specific topic. The simulator's vehicle control node subscribed to this topic uses these messages to navigate the vehicle in the virtual environment.

Figure 9 depicts the topology of the entire framework.

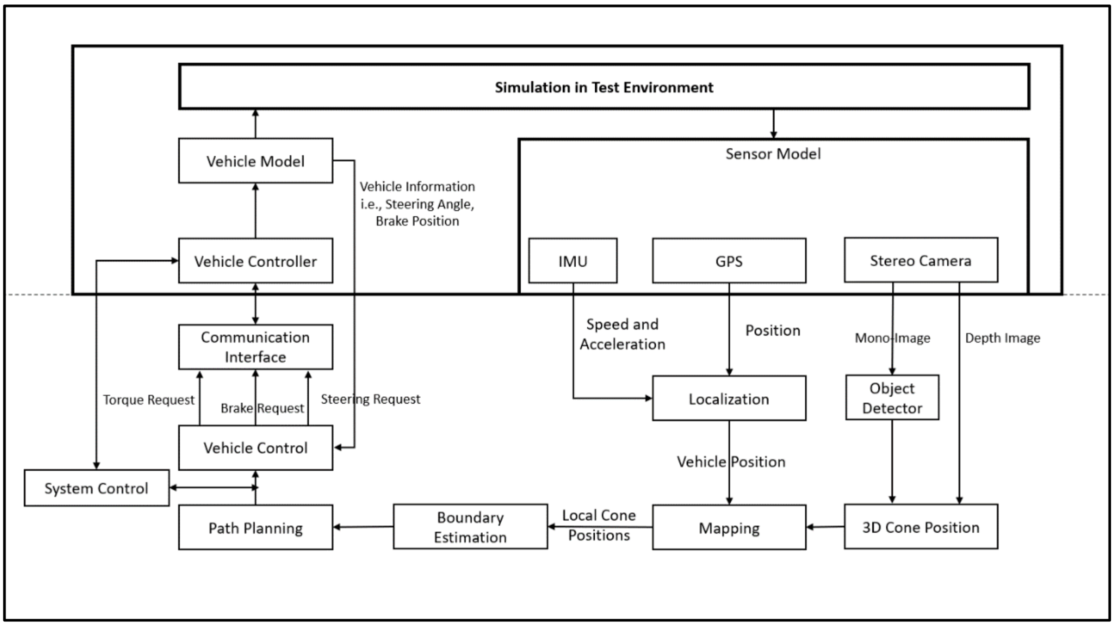


Figure 9 System Overview

The CarMaker ROS interface plugin was readily available. However, an additional adaptor was written in C++ to extract and convert the required messages into appropriate formats for the developed system's specification, allowing the two platforms to interact. Once the communications were established, the control algorithm navigated the car around the track. It made decisions about acceleration, braking, and steering based on the vehicle's position and target location.

2.3 Validation and Discussion

The virtual platform was ready for simulation after establishing the communication between the simulator and the ADS. However, before performing simulations to study the effectiveness of the simulator, it was essential to validate the developed vehicle and sensor model. Validating these models ensured that the simulation accurately represented the real-world behaviour of the car and its sensor. This was achieved by following a systematic industry-oriented approach based on a) open-loop validation and b) closed-loop validation, as discussed (Matute-Peaspan, et al., 2021).

Open-loop Validation Test

Step1: **Validation of the Vehicle Model**

A simple vehicle dynamic test using an inbuilt driver model was performed to validate the functionality of the vehicle model. This included measuring the acceleration, distance travelled, braking and steering. The obtained results were compared with the data from experimental tests.

Closed-loop Validation Test

Step2: **Validation of ADS pipeline**

Once the vehicle model was validated, the next step was to validate the developed ADS pipeline and check if proper communication had been established between the ADS and the simulation software. The vehicle was controlled using the developed motion control algorithms instead of an inbuilt driver model for this step. Besides, an ideal sensor model which gives ground truth information was utilised, disregarding real-world factors such as sensor noise. This process enabled testing and confirming if the ADS operated as intended and could drive the car in the simulation environment.

Step3: **Validation of the sensor model**

Following the validation of the ADS pipeline, the next step was to integrate and validate the custom-developed sensor model. Sensor validation is an essential phase in the simulation of AVs as they are the sole data source for the ADS to understand its state and perceive its surroundings. The more accurate the sensor models are, the more realistic testing is conducted. These custom-made sensor models were modelled considering the sensor's accuracy, noise characteristics, field of view and distortion to replicate real-world sensor behaviour. The developed sensor model was validated by comparing the simulated and real-world data using performance metrics, including object detection accuracy, distance measurements, resolution, quality of output, etc.

The required experimental test and simulation data were collected using ROS Bags to validate the simulation models.

Alternative Approach

	CURRENT APPROACH	**ALTERNATIVE APPROACH**
SIMULATOR	Vehicle Test Simulation (i.e. IPG Carmaker)	Game/Physics Engine for simulation (e.g. Unity) Robotics Simulator (e.g. Gazebo) (Rosique, et al., 2019)
VEHICLE AND SENSOR MODEL DEVELOPMENT	Parameterising, i.e. Defining requested values to model the system	Development of a highly accurate model in Simulink and integration with the simulation software

Table 5 Alternative Approach

Limitations

- In IPG Carmaker, the exact length of the vehicle could not be modelled due to the slightly higher mass, resulting in not fulfilling the dimension-to-weight criteria defined in the software. The total length of the model was extended by 51mm towards the rear end.
- The tire's wear state could not be modelled in the simulation.
- The simulation software supported only one RSI camera sensor per sensor cluster.
- The status check in the ADS-DV could not be modelled in the simulator.
- Some parameters required for modelling the suspension and powertrain model were unavailable, so highly accurate modelling could not be possible.

Assumptions

- The under-defined vehicle model (dynamics and powertrain) does not significantly influence the simulation to test and validate the ADS for this reference model, considering the low operational speed of the vehicle.
- The latency in data processing and decision-making is almost negligible.
- The minor variation in reflectivity and colour of the objects due to the material properties does not substantially influence object detection and classification. It is assumed to be the same as in the real world.
- Weather conditions in the simulation are as accurate as in the real world to test the system's performance in different scenarios.

General assumptions to be considered while simulating conventional autonomous vehicles.

- In the simulation, the motion models are idealised, i.e., the wear of dynamic components, tyre pressure, and road conditions are always considered constant or perfect.
- The behaviour of an environmental object is always predictable and deterministic in simulators, which is not the case in the real-world test. (e.g. human intentions and varying thoughts while crossing roads)
- Lighting effects in the simulation and real-world testing are the same. (In IPG carmaker, the sun position can be changed for a more realistic lighting effect).
- The static objects (buildings, road signs) and dynamic objects (pedestrians, vehicles) accurately represent and interact realistically within the virtual world.

3. Results and Discussion

3.1 Results

3.1.a Validation of Vehicle Model

Acceleration and Braking Test

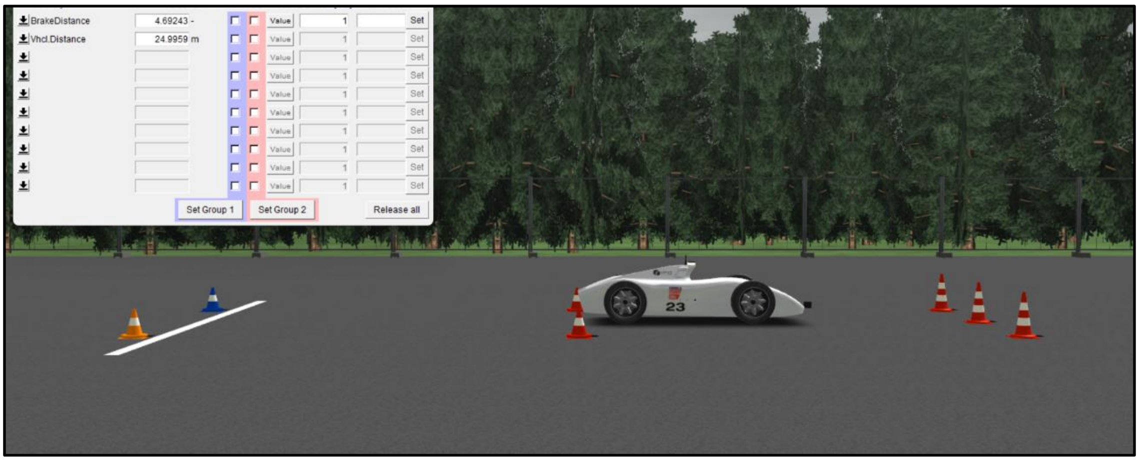


Figure 10 Straight-line Acceleration Test in IPG Carmaker

	EXPERIMENTAL TEST (AVERAGE)	IPG CARMAKER	DEVIATION
TOTAL DISTANCE	26.4 m	24.99 m	5.33 %
BRAKING DISTANCE	6.15 m	4.69 m	23.71 %
MAX SPEED ACHIEVED (NO SPEED LIMIT)	29.71 km/h	31.71 km/h	6.29 %

Table 6 Results of Straight-line Acceleration Test

Table 6 comprehensively compares experimental test results and data obtained from simulations for the conducted brake test.

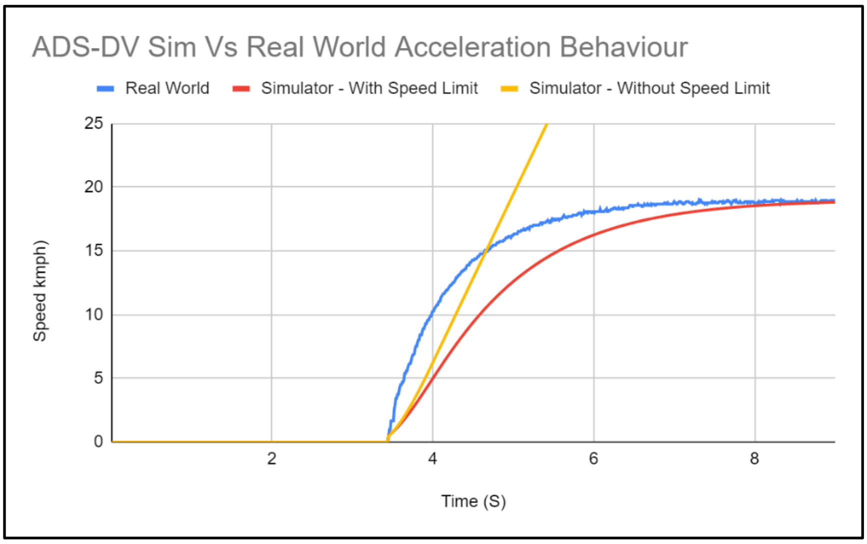


Figure 11 Acceleration Behaviour of ADS-DV in Real-world and Simulation

CRITERIA WITH SPEED LIMIT (18KM/H)	REAL WORLD	SIMULATOR-WITH SPEED LIMIT (18.9 KM/H)	SIMULATOR-WITHOUT SPEED LIMIT
MAX SPEED ACHIEVED AT (20 M)	18.9 km/h	18.97 km/h	39.46 km/h
TIME TAKEN TO ACHIEVE (18 KM/H)	5.94 secs	7.15 secs	4.89 secs
DEVIATION FROM REAL WORLD TESTING (TIME)	NA	20.37 %	17.67 %

Table 7 Results of Straight-line Acceleration test (acceleration behaviour)

The graph in ***Figure 11*** represents the vehicle acceleration performance across the real-world test and the simulation. At the same time, ***Table 7*** logs the result of the same graph in numbers for a specific point of interest.

Steering Response Test

Error! Reference source not found. compares the steering response between the simulation model and the real test in clockwise and anticlockwise directions.

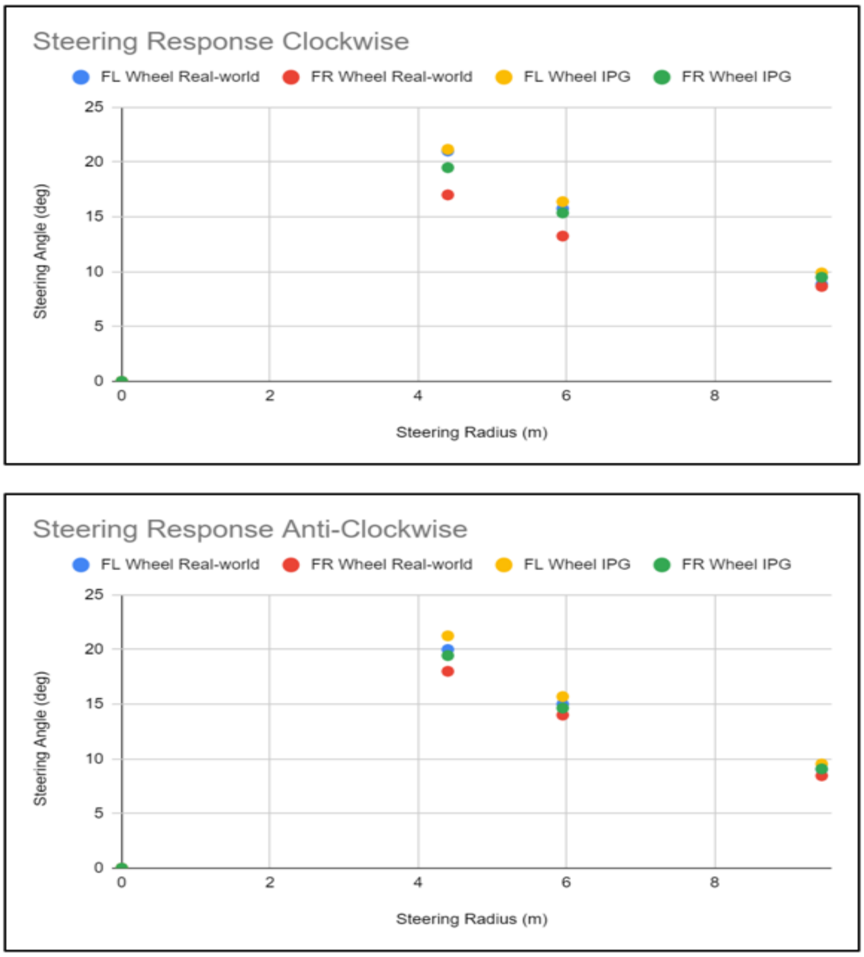


Figure 12 Steering Response of the test vehicle in Real-world and Simulation

3.1.b Validation of Control System Implementation

A graphical representation of the system architecture is shown in ***Figure 13*** using the rqt_graph, which demonstrates successful integration between the two platforms and illustrates the dynamic relationship and connection between various nodes in the running ROS system.

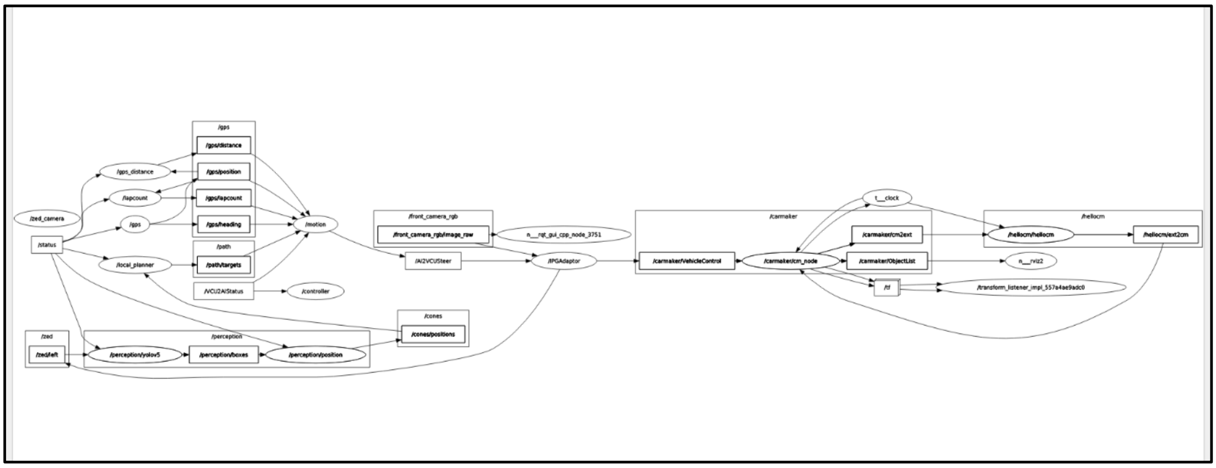


Figure 13 rqt_graph of the system architecture

Figure 14 is a snapshot from the RViz window, which illustrates the output of the camera sensor being published in the ROS network, which the software stack utilised to plan the path.

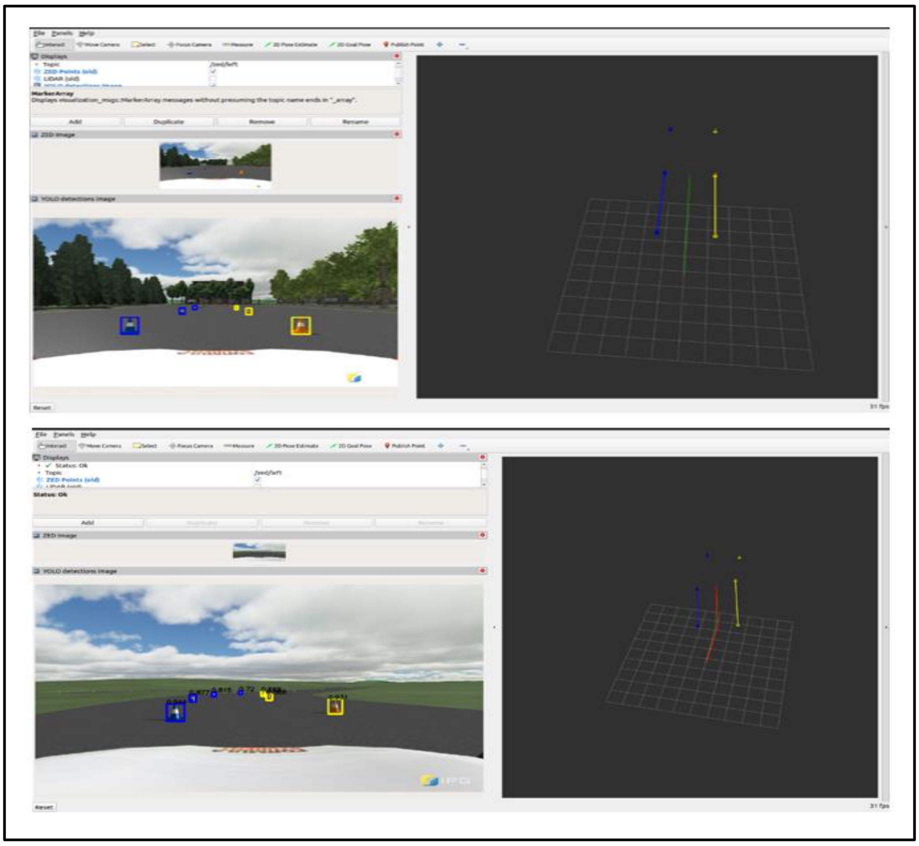


Figure 14 Object detection and Path planning based on sensor input from virtual environment suggesting successful integration of Software Stack with Simulator

3.1.c Validation of Sensor Model

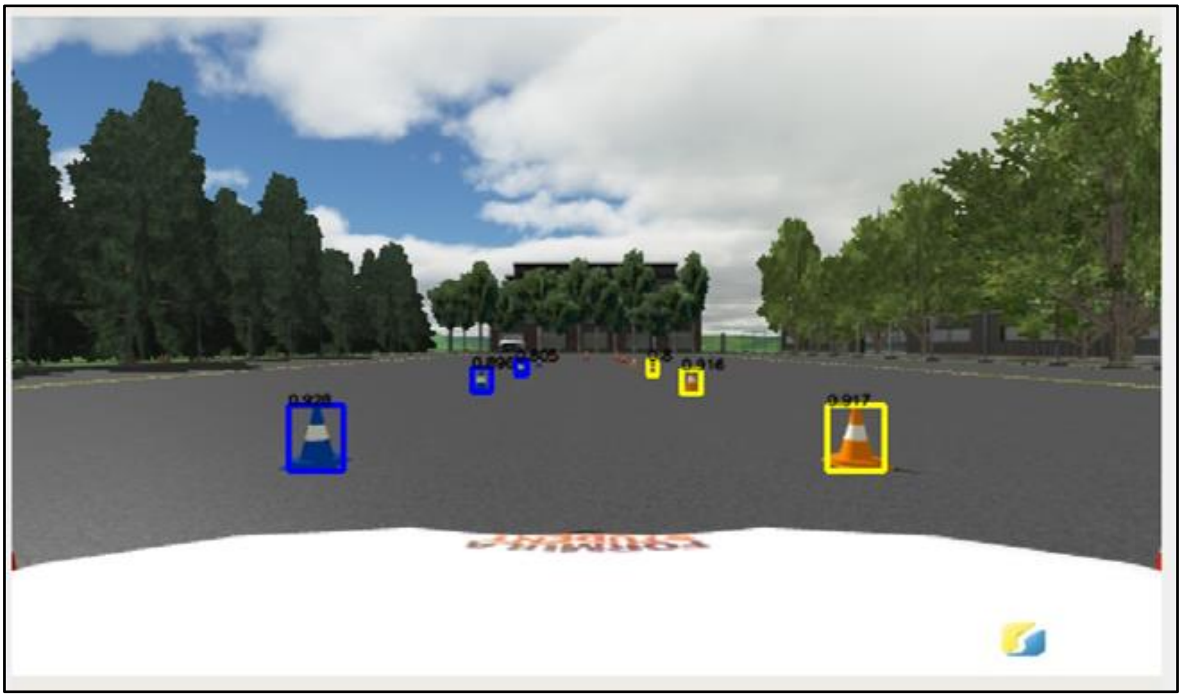


Figure 15 Simulator Sensor Output

Figure 15 shows the simulation sensor output, which was exported at the exact resolution (608*352) as in the real-world sensor to meet the requirement for the ADS pipeline.

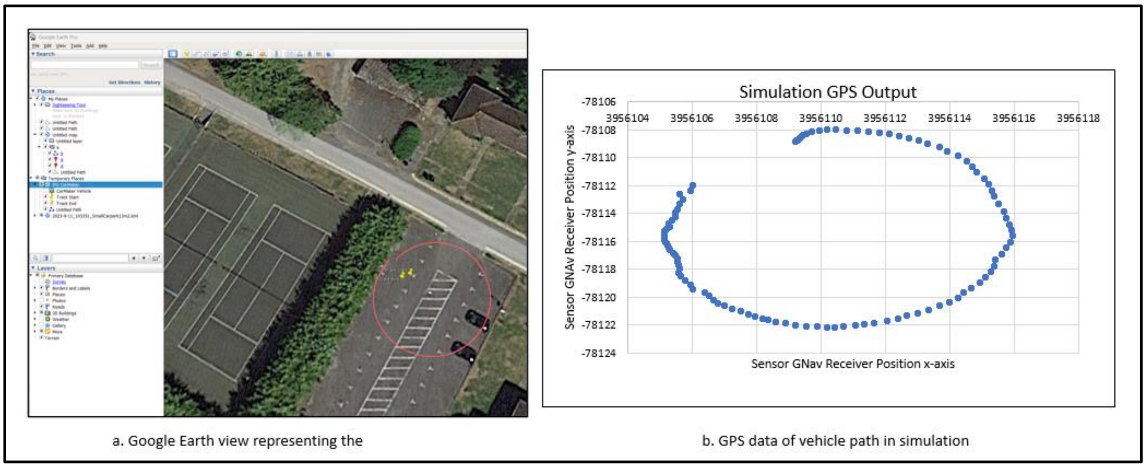


Figure 16 GPS sensor data validation with Google Earth compared with real-world data

Figure 16 shows a comparison of the GPS data from real-world and simulation using Google Earth, where a) part of the figure includes the vehicle path in the real-world test and the two yellow icons indicate the start and stop points of the simulation test, whereas the b) part shows the GPS output of the simulation vehicle.

3.1.2.a Software-in-the-Loop Simulation

a. Testing of the Perception Pipeline

Figure 17 illustrates the object detection confidence level to test the performance of the perception system simulation under varied test conditions.

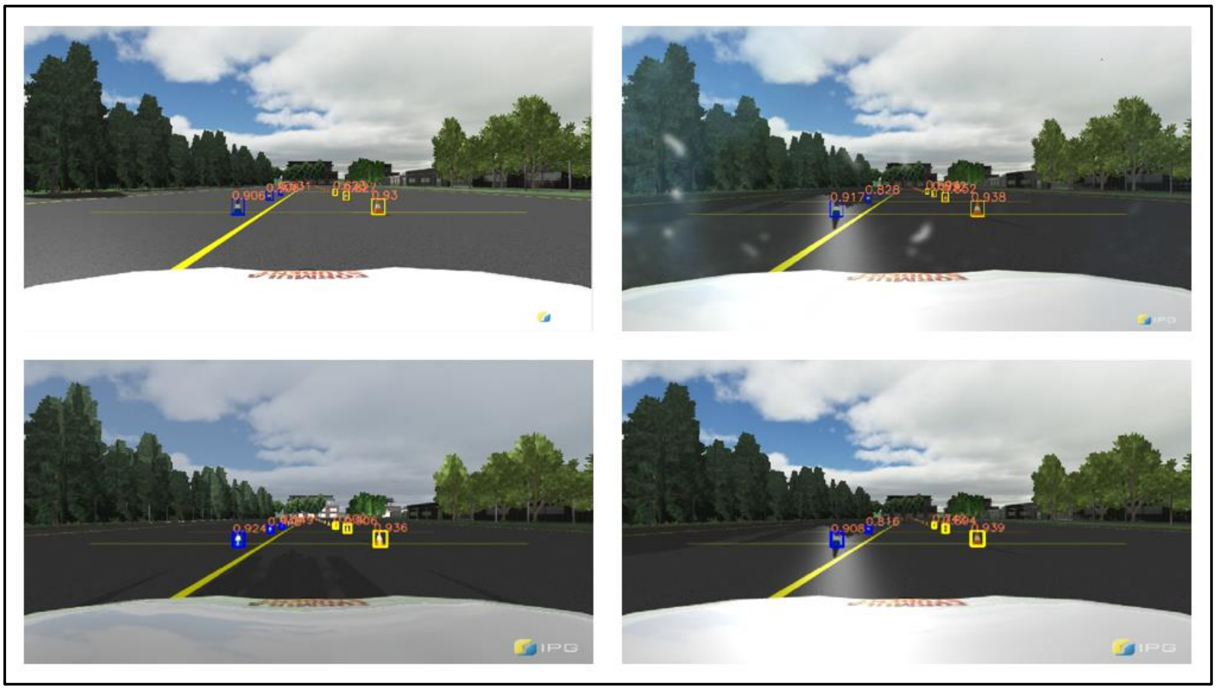


Figure 17 Perception system testing across different lighting effects (sun position) and static occlusion consideration

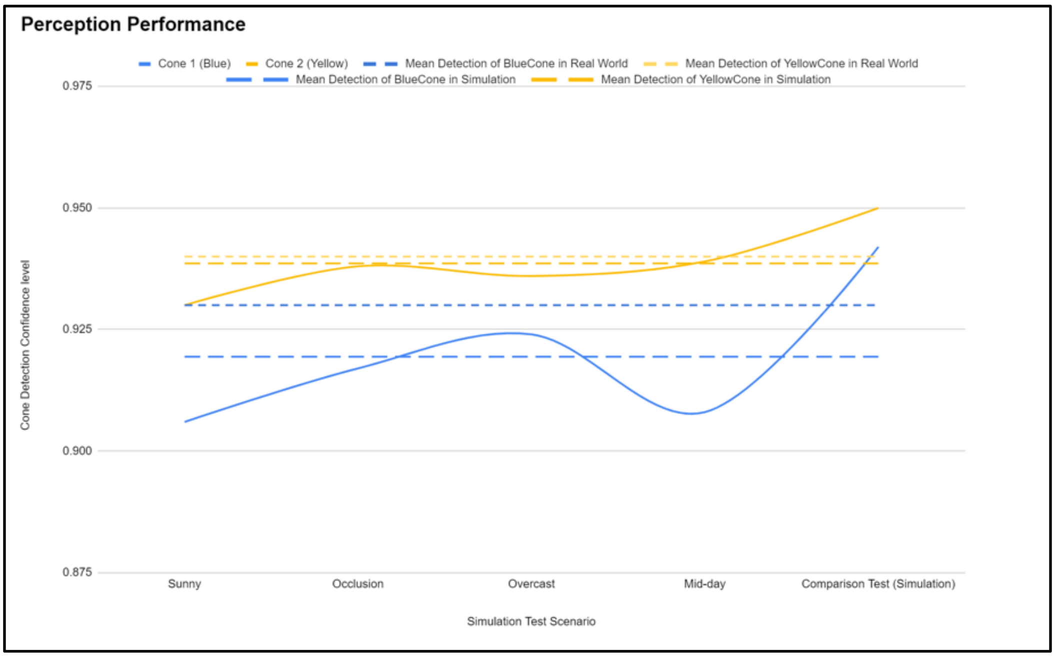


Figure 18 Perception Performance on simulation output (Influence of lighting effects and occlusion on Cone Detection)

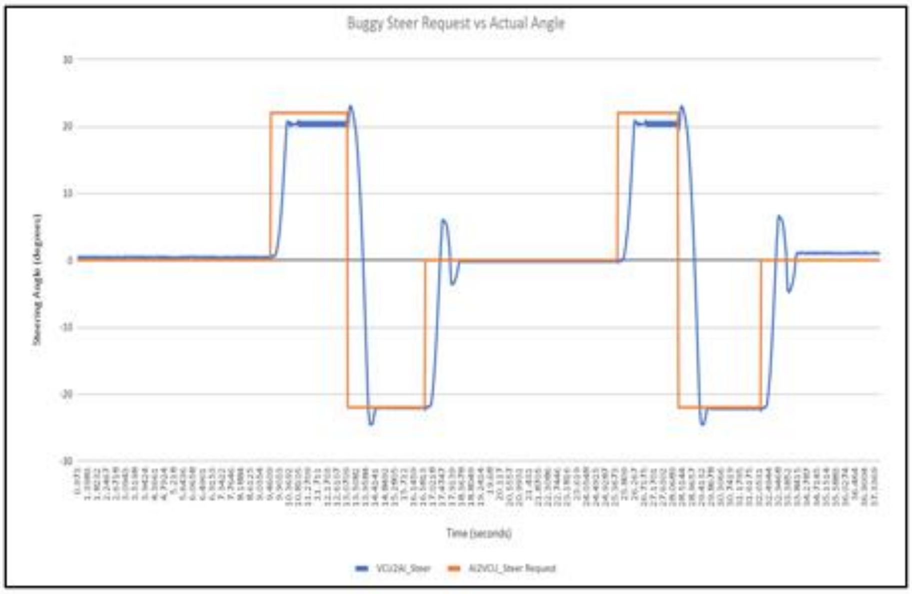
Figure 18 illustrates the performance of object detection from ***Figure 17*** in graphical form, compared with the mean detection confidence level in real-world and simulation.

Figure 19 False Positive Behaviour (Simulation and Real-world) at different locations

Figure 19 shows the False positive behaviour in the two testing environments.

3.1.2.b Hardware-in-the-Loop Simulation

a) HIL simulation performed with ADS-BV



b) Steering response of ADS_BV vs. Requested steering response

Figure 20 HIL Simulation with ADS-BV to study the response of the Steering system

3.1.2.c Vehicle-in-the-Loop Simulation

Figure 21 Vehicle-in-the-Loop Simulation of Track Drive Test in Wheatly campus car park

3.2 Discussion

3.2.1 Validation

3.2.1.a Validation of Vehicle Model (Functional Testing)

It was the first step in the simulation process; simple functionality tests were carried out to assess the vehicle dynamics accuracy in the simulation environment. The key consideration of this test was to validate how accurately the vehicle model in simulation mimics the real vehicle's behaviour.

A straight-line acceleration test was carried out as part of the longitudinal performance validation assessment. In this test, the vehicle had to accelerate from the stand-still position, travel in a straight line for 20m, and apply 100% braking. The maximum speed achieved, the time taken to travel 20m, and the braking distance was recorded and compared. This test provided valuable insights into the vehicle's responsiveness and dynamic capabilities.

In the real-world test, the vehicle was controlled using the developed ADS. The perception system was active, so the car considered the cone position to plan its path, resulting in constant minor steering adjustments. This was not the case with the simulation for this particular test, as an in-built driver model was used to control the vehicle.

The comparison indicates a marginal variation in the observed measures for total distance travelled and maximum speed achieved (less than 7% accounted for); however, a significant difference of 23.71% was observed in the braking distance between the two approaches.

While investigating for the discrepancy in the results, the following factors were observed to contribute to the variation:

1. Road Surface Conditions: The road surface conditions in both the test platforms varied in terms of traction, coefficient of friction, etc. Despite the option to alter the coefficient of friction, the simulation software generally assumes a standardised road surface, which does not entirely capture the diverse conditions encountered on real roads.
2. Tire Characteristics: Tire behaviour, including tire type, tread depth, and pressure, significantly influences braking performance. A detailed non-linear tire model was developed in the simulation software; however, the exact physical condition of the tire could not be modelled. It was observed that the vehicle's tire in the real-world had worn out, having reduced tire grip, resulting in slip and increased braking distance.
3. Vehicle Dynamics model: As simulations are based on mathematical models of the vehicle dynamics, which include weight distribution, suspension characteristics, aerodynamics, etc., the underdefined model of these parameters, due to insufficient data, were assumed to have contributed towards the difference in braking performance.

Further, an acceleration behaviour test was conducted to study the variation in maximum speed achieved. The same straight-line acceleration was performed; however, a speed limit of 18km/h was defined on this occasion. The time taken to attain the set speed was measured.

Despite the set speed limit, a maximum speed of 18.97 km/h was recorded in the real-world test. Several factors could have contributed to this variance (like instrument calibration, measurement accuracy, environmental condition, tire wear, data processing error, etc.); however, to focus on the acceleration behaviour, the attained speed of 18.97 km/h was set as the limit for the simulation test. To understand the behaviour response, the settling time and settling speed (18.972 km/h) were neglected, and the time taken to reach 18 km/h was considered.

The vehicle in the simulation was observed to have a slower acceleration rate than the real world. The same test was repeated in the simulation to investigate this delay further, but this time without any set speed limit. This iteration resulted in a significantly improved acceleration behaviour where the simulated vehicle reached the 18km/h mark in just 4.89 secs.

This comparison of results indicated the errors associated with the speed controller model. An iterative trial and error method of changing the P and I value of the speed controller was employed to close this gap; a very negligible difference was observed, the type of in-built driver model (passive and aggressive driver model) with the combination of controller model values was also tested however no considerable change in the results were observed.

Steady-state cornering tests were conducted for different radii at a constant speed for steering control validation. An increase in deviation with the steering-angle was observed. Most of the time, the findings were still considered reasonable, given that an offset in the physical vehicle's steering system was found, which contributed to this difference.

Underdefined modelling was identified as a critical contributor to the observed deviation during the functionality tests. Eventually, considering the limitations associated with the availability of vehicle parameters and the assumption made regarding the insignificant influence of the vehicle dynamic model in testing the autonomous functionality of the reference model(due to the low speed of the vehicle for the actual test run), the variation in the results were accepted to continue the study focusing on simulation of an autonomous vehicle in the virtual environment.

However, in reality, the validation process is not only about identifying discrepancies; it is also about rectifying them to improve the fidelity of the simulation. By addressing the underdefined modelling and minimising the gap between simulation and reality, the overall accuracy and reliability of the autonomous vehicle simulation can be improved.

3.2.1.b Validation of ADS pipeline

For simulating an autonomous vehicle in the virtual environment, it is essential to establish successful communication between the simulation platform and the Software stack. This integration enables the simulator to exchange data with the ADS to perform Autonomous driving based on the developed control logic. This facilitates testing and validating the ADS pipeline.

As AV functionality depends on multiple components, a typical technique to assess an ADS pipeline is observing the vehicle's adaptability to its surroundings through its path-planned driving pattern, including acceleration, deceleration, braking, steering, etc.

RViz, a 3D visualisation tool provided by ROS, was used for this purpose. It enables us to see the AV's perception of its simulation environment by creating an accurate depiction utilising the sensor data (Chunab-Rodriguez, et al., 2022). ***Figure 14*** shows that the sensor data from the simulation environment was published, and the software stack could successfully recognise the cones and plan their path. The vehicle in the virtual environment could complete a lap around the developed track as intended, validating the integration of ADS and the performance of the software stack.

3.2.1.c Validation of Sensor Model

Perception is crucial to the ADS, allowing the AV to see and interpret its environment. The pipeline for the behaviour of autonomous vehicles, i.e. perception, localisation and mapping, decision making and controls, all rely on the quality of the sensor's information (Fursa, et al., 2021). Therefore, sensor modelling and validation emerge as the most critical aspect for simulating the autonomous vehicle in the virtual environment. The quality of the sensor information in the simulation environment mainly depends on two factors: 1. Fidelity of the sensor and 2. Fidelity of the generated test environment. The fidelity of the sensor model relies on the level of detail of the sensor model, which includes Physical Characteristics (such as dimension, position, orientation, etc.), Sensor Parameters (Field of View, range, resolution, frequency, sensitivity, noise, Data formats, resolution etc.) and Dynamic effects (Occlusion, Reflections, motion blur). Whereas the fidelity of the environment depends on factors such as lighting effects, reflective index and material properties of objects, scenario diversity, etc.).

Stereo Camera Validation

CATEGORY	CAMERA
TYPE	Passive
SPECTRAL COMPONENTS	Visible light and near-infrared light
DISTANCE CAPTURING	Stereo Camera
VELOCITY CAPTURING	Optical flow
RAW DATA	Image with RGB Values
STRENGTHS	Object classification
WEAKNESS	Weather condition

Table 8 Overview of sensor properties (Schlager, et al., 2020)

Sensor noise is an essential component of sensor modelling, which refers to inherent variability, fluctuations or errors in the measurements provided by the sensor. They lead to inaccuracies in perception, object detection and localisation, leading to incorrect decisions impacting the performance and safety of the overall system. Without modelling the sensor noise, the actual performance of the autonomous systems cannot be evaluated.

Sensor noise is influenced by various factors, such as environmental conditions, manufacturing imperfections, electromagnetic interference, calibration, etc. A specific scenario was selected and recreated within the virtual environment to estimate a sensor noise value. An iterative approach of changing sensor noise values was carried out to achieve a similar resemblance to the real-world sensor output for a particular scenario. (This measures the quality of the output image in terms of noise, environment lighting and pixel distortion.)

Figure 15 shows the simulation sensor output, which was exported at the exact resolution (608*352) as in the real-world sensor to meet the requirement for the ADS pipeline.

	Depth camera	**RGB camera**
Sensor noise	0.05	0.02
Pixel saturation threshold	1.2	1.2

Table 9 Sensor Noise Modelling

After achieving a close resemblance to the real-world sensor output in terms of visualisation and confidence level in object detection, the sensor model was fixed and ready for simulation to study the effectiveness of the simulation platform.

GPS Validation

The project model uses a GPS sensor to estimate its position about the latitude and longitude to build a track map during the event. It enables the vehicle to plan its path to reduce the dependency on the perception data after finishing the first lap to complete the event faster. Besides, the GPS sensor data was also used for the lap counter to track the number of laps completed. For the GNAV sensor model in the simulation environment, the simulator allows the replication of the satellites in orbit and their visibility for the receiver vehicle. The simulator estimates the position and velocity of all the GPS satellites in their orbit using a navigation message file (daily data file uploaded by NASA) containing the parameters required to simulate the satellite's position for a given time (IPG Automotive, 2022).

A track drive test was performed in the simulation environment to verify the functioning of the GPS sensor. ***Figure 27 A6*** illustrates the GPS sensor output of the simulated vehicle path compared to the test track, demonstrating the functioning of the GPS sensor model.

In reality, the signals from the satellite are affected by several errors, including Synchronization errors of the atomic clocks, signal transmission, receiver noise, signal reflections and geometric uncertainties contributing to measurement error (Rosique, et al., 2019). The default values for noise and error parameters were considered for modelling the GPS sensor. These error models were based on the first-order Gauss-Markov process, which was parameterised using the correlation time and the standard deviation.

A simple circular test was carried out to validate the simulated GPS sensor with the actual sensor. Firstly, the test was performed in the physical world whose GPS coordinates were used for developing a track in the simulation environment, as shown in ***Figure 25 A4***. The vehicle model in the simulation performed the same test whose GPS sensor output was used for validation. The IPG carmaker software also allows integration with Google Earth to track the vehicle's real-time position with real-time speed on tracks developed using GPS data. It was observed that the simulation trace of the car in Google Earth was very accurate. Those were the exact points during the physical test.

3.2.1.d Twin environment testing

Twin environment testing, or digital twin, connects real and virtual worlds to test and optimise the system. The method involves creating a digital replica of the physical system, process and environment within the simulation platform. Real-world sensor data are integrated into the digital twin to ensure that it reflects the current state of the physical system. During the test, the vehicle's response in the two environments is recorded to observe its performance under different conditions. Any discrepancies or deviations between the digital twin's behaviour and physical system are identified and analysed, and adjustments are made to the virtual model to improve its accuracy. This approach is a process of rigorous testing and validating the virtual environment to ensure that it accurately represents the physical system, which can be further used to monitor, control and do predictive analysis.

3.2.2 Effectiveness of Simulation Software

Effective testing and validation of AVs continue to be a concern within the testing group (Mullins, et al., 2017**.** As mentioned in the journal article (Rosique, et al., 2019), conducting a series of experimental field tests to verify the functionality of an AV in a realistic timeframe is extremely difficult. In this context, simulators have become increasingly critical to close the gap to enable severe testing and timely evaluation of the AV system within a controlled environment. The simulation offers the flexibility to perform rapid iteration, allowing to test and validate the performance of the developed autonomous system under diverse scenarios. Simulation software has emerged as a pivotal tool in this endeavour, offering multiple approaches like Model-In-the-Loop(MIL), Software-In-the-Loop(SIL), Hardware-in-the-Loop(HIL), and Vehicle-In-the-Loop simulation(VIL). These techniques allow the testing of actual hardware components and conceptual models of the AVs.

The V-Model technique, a systematic approach followed in the automotive industry for software-based development and testing of autonomous vehicles (Rajabli, et al., 2020), was employed in this section to explore the effectiveness of simulation software.

3.2.2.a Software-in-the-Loop Simulation

The SIL simulation gives a tactical advantage in identifying potential flaws in the system's pipeline. It allows for the execution of control unit code through ROS coupling before hardware is available, allowing for testing of the entire pipeline, including software components and simulated software models of the AV.

SIL testing also permits isolating specific components or subsystems, making it easier to identify and fix issues of individual software modules without interfering with other elements.

Example:

a. **Testing of the Perception Pipeline**

The perception system is an essential component of the ADS, and as stated in (Fursa, et al., 2021)**,** different weather conditions can significantly impair perception. Since the camera sensor belongs to the passive sensor type dependent on the external lighting source (Schlager, et al., 2020), these sensors are greatly influenced by the lighting condition. If this is not considered, the subsequent AV control system pipeline may be exposed to inaccurate or incomplete testing. Because of this, AV technology must be developed and tested extensively in all possible settings and weather situations.

Modifying the simulation to evaluate the perception performance under various scenarios and conditions, from extreme weather to rare-edge instances, is simple. This scalability is difficult to achieve with physical testing. ***Figure 17*** demonstrates the performance perception system performs in several test settings, considering the lighting and occlusion texture impacts. A noticeable change in the confidence level of object detection was recorded in each test case, suggesting the influence of environmental and physical aspects on perception. The perception performance results agree with the result pattern in (Fursa, et al., 2021), where the confidence level for object detection from simulator output is lower than in the real world. However, a conducted test with a close resemblance in terms of overall lighting effect showed a good agreement with the real-world test result. This demonstrates the ability of the simulation software to test the perception system within a controlled environment.

b. **False positive in Object Detection**

False positive refers to a situation when the system identifies an object that does not exist or misclassifies an object. In the context of autonomous vehicles, false positives can lead to incorrect decisions, affecting the path-planning process.

In an event, while conducting a real-world test in the Wheatley campus park area, it was observed that the vehicle was going outside of the track on specific instances. Upon investigation, it was regarded as an outcome of a false positive where the ADS mistook yellow markings on the road as cones, leading to incorrect path planning and forcing the vehicle to go outside the track. The same behaviour was observed in the simulation environment, as seen in ***Figure 19***. Although the false positive was not located in the same spot, meaning that object detections within are randomly confused, its conduct was comparable, demonstrating the usefulness of using the simulator to anticipate prospective problems.

c. Optimization of Motion Control Pipeline

This testing approach helps estimate the performance of the software stack to plan a path and the subsequent control action for navigating the vehicle autonomously. By observing the vehicle's behaviour in simulation, the developers can fine-tune the control logic to meet the expected outcome. During the real-world test, the car was observed to take a longer path during cornering, knocking a few cones, which depended on multiple aspects, including the look-ahead distance and the steering controller. To achieve an appropriate response for the vehicle, repeated tests were conducted with iterative changes being implemented, which was a time-consuming process. However, performing the same iteration in the virtual environment would allow for more effective testing, enhancing the overall development and deployment process. Similarly, during the SkidPad event in the competition, the vehicle was observed to take the same path and hit the same cone as in the simulation(gazebo simulator) since this behaviour was already predicted and diagnosed as an outcome of incorrect steering angle (confusion between "–"ve and "+" ve sign for clockwise and anti-clockwise steering), the changes were implemented to the physical system. Such accurate prediction of the vehicle behaviour increases the reliability of the simulator.

In brief, the SIL simulation is a very beneficial method that allows testing various parameters and functionalities of the autonomous vehicle like sensor fusion, localisation and mapping, Path-Planning and control, object detection and tracking, behaviour prediction, etc., in a controlled environment allowing necessary changes to be implemented advancing the overall development process.

3.2.2.b Hardware-in-the-Loop Simulation

Hardware-in-loop simulation integrates the physical hardware system or component with the virtual model to create a closed-loop testing environment. This technique bridges the gap between virtual and real-world assessment. It is helpful to test and validate the performance and functionality of the specific hardware system, enhancing the authenticity and effectiveness of testing in a controlled and comprehensive manner.

HIL simulation was performed in this project with the a. In-car PC in loop b. Steering unit of ADS-BV as the hardware system in the loop.

a. In-car PC in the Loop test

In-car PC is the brain of the ADS-DV, and it contains the developed software stack, including scripts for object detection, object classification, distance measurement, path planning, controls, etc. Performing HIL with an In-car PC allows testing and validating the system before the vehicle is available. This trial also facilitated testing the interactions/ networks of the components within the virtual prototype.

b. Steering in Loop Test

This simulation-based test evaluates the performance of the vehicle's steering system in a controlled environment. In this test, ADS-BV was used due to the limited availability of the considered reference vehicle. Despite being a different model, the ADS-BV was used to perform the HIL test to study and demonstrate the effectiveness of the simulator.

Figure 20 shows the behaviour of the actual steering angle compared to the requested steering input. This approach offered several advantages to contribute to developing and validating the steering system. It was helpful to study the deviation in the actual steering angle in response to the requested steering angle, the delay/time taken to achieve the requested response, etc. Using this HIL approach, the steering response delay of the physical vehicle was lowered.

Critical benefits of the HIL test include Safety, Cost Efficiency, Stage iterations, Scenario Replications, Sensor and Actuator Integration, Data Collection, Closed Testing (Interaction with other sub-systems), algorithm validation, system calibration, etc.

3.2.2.c Vehicle-in-the-Loop Simulation

The VIL simulation test offers an advantage to test the control of ADS on the actual vehicle without the requirement for reproducing the test track in the real world. In this approach, the real-test car was embedded in the virtual environment, where information from the virtual sensor model was injected into the real-world system to test the vehicle behaviour triggering the steering and braking actuators. ***Figure 21.*** Illustrates the VIL simulation for the track drive test performed with the ADS-BV. Sensor and status control messages from the simulation PC were sent to the In-car PC over the CAN network.

3.2.2.d Report Findings

Simulation Software Performance

1. Accuracy and Realism: Though specific details of the real-world testing could not be perfectly modelled in the simulator, the simulation software demonstrated a commendable level of accuracy and realism in replicating the real-world test scenario. In the tests conducted, it successfully mimicked various environmental factors, including the required infrastructure (cones) and dynamic lightning effects (sun position), with a high degree of fidelity.
2. Customizability: The software’s customizability was a significant advantage. It allowed for easy parameter modification, enabling diverse test scenarios to adapt to specific study requirements. This flexibility was valuable in assessing the behaviour of the autonomous vehicle across various conditions.
3. Runtime Efficiency: One of the critical factors in the simulation software is its runtime efficiency. The software performed efficiently, providing real-time simulation feedback, even with complex scenarios. This efficiency significantly reduces the time required for testing and data collection.
4. Validation against Real-world Data: To validate the accuracy of the simulation software, the obtained test results were compared with real-world data from the test conducted on the university campus and during the event. The software consistently

produced results that matched real-world data. The false positive detection and the similar driving behaviour compared to the real-world test indicated the capability to replicate actual driving conditions effectively.

The study's findings highlight the significance of the simulation platform in developing and testing autonomous vehicles. The software's accuracy, customizability, runtime efficiency, and flexibility in testing different components of the systems make it a powerful tool for the development team. Additionally, the simulation behaviour's near resemblance to actual trials gave rise to trust in its effectiveness.

3.2.3 Simulation Software Enhancements

Considering the observations recorded during this project, it can be interpreted that the simulation platform's effectiveness for testing and validating AVs is mainly influenced by the realism of the simulation environment, which can be classified into Physics Realism and Environmental Realism. The simulation's physics realism is determined by how well it mimics actual physical principles, such as the sensor behaviour, vehicle dynamics, and environmental conditions. The fidelity of the subsequent models influences the realism of simulation; therefore, it is critical to develop a highly accurate model to assess the true behaviour of the concept.

The flexibility to integrate models developed using dedicated simulation tools will allow the development of a more reliable platform. For example, IPG carmaker offers the flexibility to incorporate vehicle dynamics models created using third-party software like MATLAB/Simulink and powertrain models developed using GT-suite or AVL Cruise, etc. Such precise modelling and integration of specific components improves the overall accuracy of the physics simulation.

Similarly, the environmental realism represented by the type of road, weather conditions, lighting effects and, importantly, the realistic visuals affect the overall performance of the simulation. Companies like Atlatec and rfpro are developing high-precision 3D maps of real-world environments using Camera, Lidar and GPS data, which can be imported into simulation software for creating test scenarios for ADAS and Autonomous applications. Also, with the increasing demand for highly realistic visuals, simulator developers tend to use game engines to improve the graphics and visuals to enhance the simulation capabilities. The MovieNx, a secondary visualisation platform for IPG Carmaker, is currently being developed, focused on rendering quality using UNIGINE 2.

More realistic geometric objects (like real-world street furniture, animals, pedestrians, etc.) within the simulation software would allow easy incorporation within the test scenario, enabling testing with multiple object considerations.

4. Conclusion and Future Work

4.1 Conclusion

This dissertation has explored the topic of simulating an autonomous vehicle in a virtual environment with the specific aim of creating high-fidelity models to study and evaluate the effectiveness of the simulation platform. The project's significance lies in its potential to accelerate autonomous vehicle technology's overall testing and development.

This study yields the following findings.

- Importance of consideration of literature for software selection.
- Significance of high-fidelity modelling and model validation to enhance testing accuracy.
- The ability of the simulation software to create extensive test scenarios allows the developers to rapidly test, iterate, fine-tune and validate various aspects of AV technology. This includes both Software and Hardware.
- This approach enhances the safety and reliability of autonomous system testing and reduces the overall testing and development cost.
- The simulation software could replicate real-world behaviour, accounting for less than 3% error in object detection and almost negligible error in control behaviour patterns.

Even if achieving absolute realism in the virtual environment remains an ongoing pursuit, the conducted validation experiments and test cases in simulation closely resembled the real-world test observations. Through cone detection performance, false positive alerts, path-planning and control behaviour, the reliability of the simulation tool was affirmed. The ability to integrate hardware components to perform extensive tests within a controlled environment expands the simulation software's applications. A discussion on current developments for the enhancement of the simulator is also presented in this project.

In conclusion, this dissertation emphasises the crucial role of high-fidelity simulation models in advancing the testing of autonomous vehicle technology. The study affirms the effectiveness of the simulation software for AV evaluation. Its ability to replicate and reproduce real-world scenarios with similar behaviour makes it a valuable resource in developing autonomous vehicle testing.

4.2 Future Work

With this project, a base platform is now available, and the following listed areas can be worked open to address additional factors to conduct a thorough study for the future.

- Advancement in Test environment modelling:
  - Use real-world lidar sensor data to recreate an extremely accurate test environment within the simulation.
- Advanced Sensor Models: -
  - Develop and implement a Lidar sensor within the virtual environment and test its competence.

    - Explore using synthetic data and domain adaptation techniques to enhance sensor perception in diverse environments.
    - Testing Sensor fusion and Sensor cluster
- Advanced Vehicle model: -
    - A more accurate model can be developed using Simulink and integrated with the simulation software.
- Performing Twin environment testing with the ADS.
- Focusing on Real-world passenger cars by considering the implementation of traffic objects.

References

Aparow, V. R. et al., 2019. A comprehensive Simulation Platform for Testing Autonomous Vehicle in 3D virtual environment. *2019 IEEE 5th International Conference on Mechatronics System and Robots.*

Bezai, N. E. et al., 2021. Future cities and autonomous vehicles: analysis of the barriers to full adoption. *Energy and Built Environment,* 2(1), pp. 65-81.

Choi, H., Crump, C. & Duriez, C., 2021. On the use of simulation in robotics:Opportunities, challenges and suggestions for moving forward. *Proceedings of the National Academy of Sciences (PNAS).*

Chunab-Rodriguez, M. A. et al., 2022. A Free Simulation Environment Based on ROS for Teaching Autonomous Vehicle Navigation Algorithms. *Applied Sciences.*

Cime, K. M. et al., 2021. Assessing the Access to Jobs by Shared Autonomous Vehicles in Marysville, Ohio: Modeling, Simulating and Validating. *SAE Int. J. Advances & Curr. Prac. in Mobility.*

Dona, R. & Ciuffo, B., 2022. Virtual Testing of Automated Driving Systems. A survey on Validation Methods. *IEEE Access.*

Esteller, E. G., 2020. *Autonomous Software-In-The-Loop Modelling, Validation and Assessment,* Oxford: Oxford Brookes University.

Fursa, I. et al., 2021. Worsening Perception: Real-time Degradation of Autonomous Vehicle Perception Performance for Simulation of Adverse Weather Conditions.

IPG Automotive , n.d. *User's Guide Version - CarMaker v11.0.1.* s.l.:IPG Automotive.

IPG Automotive, 2022. *Reference Manual v11.0.1.* s.l.:IPG Automotive.

Kato, S. et al., 2015. An Open Approach To Autonomous Vehicles. *IEE Micro.*

Kim, B., Kashiba , Y., Dai, S. & Shiraishi, S., 2017. Testing Autonomous Vehicle Software in the Virtual Prototyping Environment. *IEEE Embedded Systems Letter,* Volume 9.

Matute-Peaspan, J. A., Zubizarreta-Pico, A. & Diaz-Briceno, S. E., 2021. A vehicle Simulation Model and Automated Driving Features Validation for Low-Speed High Automation Applications. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS,* Volume 22.

Muktadir, G. M. et al., 2022. Procedural Generation of High-Defination Road Networks for Autonomous Vehicle Testing and Traffic Simulations. *SAE International Journal of CAV.*

Mullins, G. E., Stankiewicz, P. G. & Gupta, S. K., 2017. Automated Generation of Diverse and Challenging Scenarios and Evaluation of Autonomous Vehicles. *IEEE International Conference on Robotics and Automation (ICRA).*

Platt, J. & Ricks, K., 2022. Comparitive Analysis of ROS-Unity 3D and ROS-Gazebo for Mobile Ground Robot Simulation. *Journal of Intelligent & Robotic Systems.*

Quigley, M., Gerkey, B. & Smart, W. D., 2015. *Programming Robots with ROS.* s.l.:s.n.

Rajabli, N., Flammini, F., Nardone, R. & Vittorini, V., 2020. Software Verification and Validation of Safe Autonomous Cars: A Systematic Literature Review. *IEEE Access, 9: 4797-4819.*

Riedmaier, S. et al., 2021. Model Validation and Scenario Selection for Virtual-Based Homologation of Automated Vehicles. *MPDI applied sciences.*

ROS, 2022. *ROS.org.* [Online]
Available at: http://wiki.ros.org/Bags

Rosique, F., Navarro, P. J., Fernandez, C. & Padilla, A., 2019. A Systematic Reviw of Perception System and Simulators for Autonomous Vehicle Research. *Sensors.*

Schlager, B., Muckenhuber, S., Schmidt, S. & Holzer, H., 2020. State-of-the-Art Sensor Models for Virtual Testing of Advanced Driver Assistance Systems/ Autonomous Driving Function. *SAE International Journal of Connected and Automated Vehicles.*

Shah, S., Dey, D., Lovett, C. & Kapoor, A., 2017. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles. *Field and Service Robotics.*

Shao, Y. et al., 2022. Real-Sim Interface: Enabling Multi-resolution Simulation and X-in-the-Loop Development for Connected and Automated Vehicles. *Special Issue on Emerging Simulation Tools and Technologies for Testing and Evaluating Connected .*

Siegwart, R., Nourbakhsh, I. R. & Scaramuzza, D., 2011. Mobile Robot Localization. In: *Introduction to Autonomous Mobile Robots, Second Edition.* s.l.:MIT Press.

Teng, S. et al., 2021. Motion Planning for Autonomous Driving: The State of the Art and Future Perspectives. *IEEE TRANSACTIONS ON INTELLIGENT VEHICLES.*

Wu, Z. et al., 2020. Research and Simulation of Autonomous Navigation for Method for Unmanned Vehicle Based on ROS. *SAE Technical Paper.*

Appendices

A1. Introduction

A1.1 Background

FS-AI UK is an International student competition organised by IMechE where teams from different countries compete against each other. As a competition component, the participating teams are expected to develop an autonomous driving system (Software Stack) integrated into the test vehicle provided by competition organisers to complete a series of events successfully. As a part of the development process, the organisers encouraged the use of simulation software, which was evaluated during the software development presentation under the statics section.

A1.2 Literature Review

In recent years, the development and advancement of autonomous vehicle technology has been a promising area of research. The technology is expected to revolutionise transportation by making travel safer, more efficient, and more convenient. However, a fully functioning self-driving vehicle is still further from reality, considering the lack of available infrastructure and the complexity involved in them. As the complexity of modern cars continues to increase, the effort involved in developing the exact increases simultaneously. Moreover, it is not easy to test and validate the behaviour of the autonomous vehicle, considering the high volume of relevant test cases and the randomness of the event using physical prototypes. The testing and validation involve high cost, human resources and a time-consuming process. The development of a simulator thus has emerged as a powerful tool for conducting tests and evaluating the vehicle's performance, which can validate the control algorithms and reduce the complexity involved. (Choi, et al., 2021)

Simulation is a state-of-the-art development in Engineering, as it enables the testing and validation of a designed model for real-world test conditions by imitating the working conditions and estimating its performance even before its manufacturing. This is advantageous in terms of cost, time and many other parameters. They are also valuable for optimising the process besides testing and validation. Simulation in a virtual environment has several advantages over physical testing; it allows tests to be conducted in a controlled environment, and most importantly, a series of repeatable and accurate tests can be carried out. This enables the developers/researchers across different departments to evaluate the vehicle's performance under various test conditions. They also make testing safe by eliminating the risk of accidents and fatalities associated with dynamic testing. The main elements for testing ADAS and autonomous driving function in a virtual prototype are the interaction between the system and the environment. The simulation requires detailed parametrisation of models (vehicle and sub-components) high fidelity realistic test environments, which the modelled sensors can capture. The simulation software provides tools for simulating vehicle behaviour and testing algorithms. The simulations can be divided into two categories: 1. Physical base simulation helps test the physical interactions of the vehicle with the environment, i.e. to verify the ability of the sensor models to detect obstacles. 2. Behaviour-based simulations help test test vehicles' decision-making algorithms.

Advantages of Simulation of Autonomous Vehicles in Virtual Environment

Cost-effectiveness: The cost required to conduct physical testing is higher and can be reduced to a greater extent by the virtual prototyping strategy. Also, in case of any failure, the cost involved in the physical tends to increase.

Safety: In physical testing, safety is at risk due to the predictable behaviour of the vehicle.

Repeatability: Using simulation software to perform the same iteration multiple times without any margin of error is a significant advantage.

Scalability

Time-Efficient: Conducting a physical test is very time-consuming, whereas using a simulator, we can perform multiple tests simultaneously without compromising test results.

Controlled environment: A physical test requires consideration of environmental factors such as weather, sunlight, road conditions, etc. This can be avoided as we can simulate and conduct tests in required test conditions.

Data Collection: Dynamic testing requires multiple sensors, data loggers, cables and other instruments to collect data. This also increases the cost, and we can model these sensors in the simulator and acquire the necessary data without any hazel.

Challenges involved in the Simulation of Autonomous Vehicles in a Virtual environment

Accuracy: The accuracy of the results obtained from the simulation majorly depends on how accurately the sensor and environment are modelled. The possibility of developing an accurate model depends on the simulator type.

Realism: Creating a realistic test condition with the minimum assumption is essential for accurate close results.

Validation: Acquiring data from physical tests conducted with the same test condition is essential to validate the result.

Computing: Simulation software performs a series of mathematical iterations and, therefore, has a high computing requirement to perform multiple iterations to estimate the output results. Also, to validate sensor models, it is essential to generate high-resolution simulations, which require better graphics.

The simulator enables the option of performing Model-in-loop (MIL), Software-in-loop (SIL), and Hardwar-in-loop (HIL) simulations, which are commonly used in the automotive industry to simulate the behaviour of vehicles and sub-systems during different development phases. This enables engineers to validate their design and control algorithms before building prototypes, reducing costs and accelerating development.

MIL simulation can be used to test a model and algorithm in the virtual environment and helps identify the issues in earlier stages of development. The SIL simulation tests the software components in a simulated environment before integrating them into hardware. In SIL, the software, the input and output signals are simulated. In HIL simulation, a physical component is integrated into the modelled environment to validate the component's functionality. This can include a physical ECU, sensors, etc.

Apart from MIL, SIL and HIL, Vehicle-in-loop is becoming popular as it allows an entire vehicle to be integrated with the simulator and tested in a controlled environment such as a laboratory, reducing the need to go out on the test track.

In his research paper, Morando discussed the importance of the simulation-based approach and its usefulness for assessing the safety influences of autonomous vehicles and addressed the limitations associated with the approach. The paper also critically discussed the software's ability to replicate the autonomous vehicle's behaviour.(Morando, et al., 2018)

To generate a high-fidelity environment for virtual prototyping, prominent research and development activities are conducted to design HD maps that provide detailed environment data and can be integrated with simulation software, enabling the testing and validation of the sensor models. In (Schlager, et al., 2020), the author discussed and summarised the characteristics of physical measurement principles and sensors to understand sensor and their operating principles. The main reason for variation in the sensor output in real-world and simulation tests is the lack of consideration of sensor effect and noise.

For interpretation of the behaviour of autonomous vehicles in the virtual environment, it is essential to understand the set of sensor models used in the car, the ability of the sensor to provide accurate measurements, the developed control algorithm and most importantly, the ability of the simulator to generate realistic 3-D environment which can be detected and interpreted by the sensor to provide necessary information to the control system so as it can decide on the action that needs to be performed.

Sensor fusion and Integration are two significant factors affecting an autonomous vehicle's performance. Current research on simulating autonomous vehicles in virtual environments involves testing the performance under a wide range of weather and road conditions and also a variety of traffic patterns, the interactions between humans and autonomous vehicles for the development of safe and efficient AV, to train the vehicle AVs and control system.

The accuracy of the simulation environment must be verified before using the simulation software as a virtual validation tool. The biggest challenge associated with the current simulation is its validation, ensuring the simulation-generated output is characterised by fidelity level (Dona & Ciuffo, 2022). The simulation modelling has its limitations concerning the software. Also, the development of real-world scenarios in the simulator is not precise since the modelling involves assumptions for parametrising the models, due to which they cannot replicate the actual test data but are considered adequate and reliable enough to be verified.

Apart from considering the fidelity of models, the other important aspect of the autonomous vehicle is the navigation method. A synchronous positioning map construction and path planning are critical parameters to achieve autonomous navigation. Robot Operating

System(ROS), a development platform, is used along with the simulator to combine mapping and path-planning algorithms to achieve autonomous navigation. (Wu, et al., 2020)

To conduct this study, an autonomous vehicle model and subsequent control code were needed; therefore, the formula student autonomous concept was considered. The OBRA team has used Gazebo with ROS interface as the simulation framework to aid their testing and development. However, the team is currently looking for an alternate solution due to certain limitations with simulation software. These limitations were considered while selecting a suitable simulation software for this study.

Based on the literature survey, this research project initially focused on comparing popular simulation software and game engines to discuss parameters required for generating fidelity environments and subsequent modelling to perform virtual testing. A comprehensive simulator development was planned to achieve the project's objective.

A2. Methodology

A2.1 Module Design, Development and Simulation

Formula Student – ADS-DV - Specification

This document contains the Vehicle Specification for the Formula Student Autonomous Driving System Dedicated Vehicle FS ADS-DV.

The Vehicle specification is as follows:

Drive System:	
Drive motors	Saietta 119R-68 (x2)
Peak motor torque	55.7Nm
Peak motor power @ 48V	17.0 kW
Continuous motor torque	27.2 Nm
Continuous motor power @ 48V	8.8 kW
Maximum motor speed	4,000 rpm
Belt drive ratio	3.5:1
Peak motor current	400 A
Motor controllers	Sevcon Gen4 DC (x2)
Operating voltage range	31-90 V
Peak current rating	550 A
Steering System:	
Steering Motor	Shiftec ZZ011-1
Steering Controller	Shiftec ZE102-1
Traction Battery:	
Cells	CALB CA100AHA
Cell chemistry	Lithium iron phosphate ($LiFePO_4$)
Cell capacity rating	100 Ah
Pack configuration	16S1P
Nominal pack energy capacity	5.1 kWh
Nominal pack voltage	51.2 V
Operating voltage range	41.6-58.4 V
Peak discharge current	800 A (up to 10 secs)
Continuous discharge current	300 A
BMS	Proprietary Hypermotive BMS
Charger	TC Charger 3.3 kW (HK-J-H66-40)

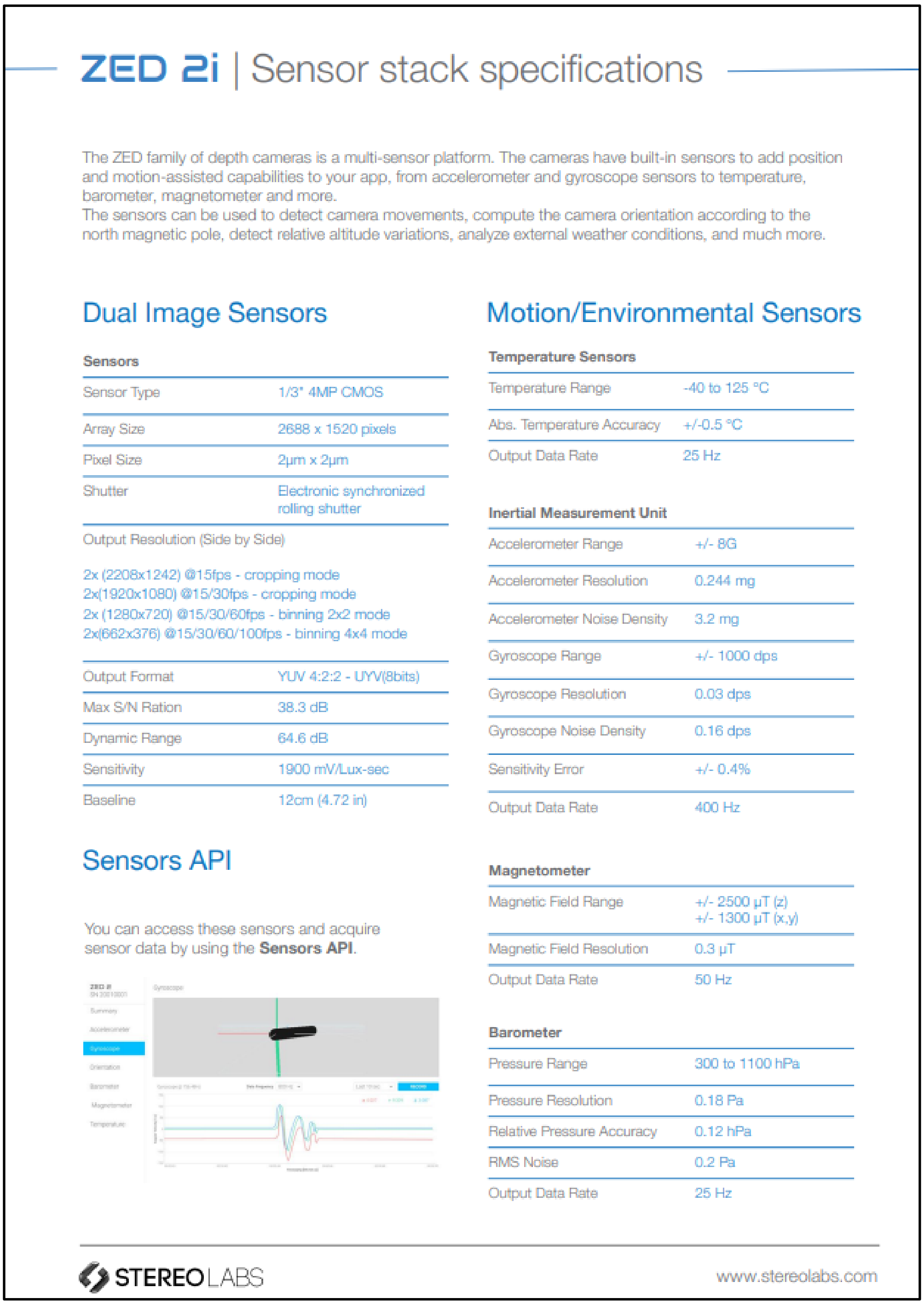
The vehicle dimensions can be found here:
Formula Student ADS_DV Dimensions and Locations

The vehicle has the following space available for an AI computer to be installed: TBD

The vehicle is provided with an umbilical connection to the AI computer which includes the CANBus communication which is detailed in the ADS-DV Software Interface Specification and power for the AI computer and AI sensors.

The connector and pin-out information is located in the ADV-DV AI-Interface - Electrical Document

Figure 22 AF1. ADS-DV Specification



ZED 2i | Sensor stack specifications

The ZED family of depth cameras is a multi-sensor platform. The cameras have built-in sensors to add position and motion-assisted capabilities to your app, from accelerometer and gyroscope sensors to temperature, barometer, magnetometer and more.
The sensors can be used to detect camera movements, compute the camera orientation according to the north magnetic pole, detect relative altitude variations, analyze external weather conditions, and much more.

Dual Image Sensors

Sensors	
Sensor Type	1/3" 4MP CMOS
Array Size	2688 x 1520 pixels
Pixel Size	2μm x 2μm
Shutter	Electronic synchronized rolling shutter

Output Resolution (Side by Side)

2x (2208x1242) @15fps - cropping mode
2x(1920x1080) @15/30fps - cropping mode
2x (1280x720) @15/30/60fps - binning 2x2 mode
2x(662x376) @15/30/60/100fps - binning 4x4 mode

Output Format	YUV 4:2:2 - UYV(8bits)
Max S/N Ration	38.3 dB
Dynamic Range	64.6 dB
Sensitivity	1900 mV/Lux-sec
Baseline	12cm (4.72 in)

Sensors API

You can access these sensors and acquire sensor data by using the **Sensors API**.

Motion/Environmental Sensors

Temperature Sensors	
Temperature Range	-40 to 125 °C
Abs. Temperature Accuracy	+/-0.5 °C
Output Data Rate	25 Hz

Inertial Measurement Unit	
Accelerometer Range	+/- 8G
Accelerometer Resolution	0.244 mg
Accelerometer Noise Density	3.2 mg
Gyroscope Range	+/- 1000 dps
Gyroscope Resolution	0.03 dps
Gyroscope Noise Density	0.16 dps
Sensitivity Error	+/- 0.4%
Output Data Rate	400 Hz

Magnetometer	
Magnetic Field Range	+/- 2500 μT (z) +/- 1300 μT (x,y)
Magnetic Field Resolution	0.3 μT
Output Data Rate	50 Hz

Barometer	
Pressure Range	300 to 1100 hPa
Pressure Resolution	0.18 Pa
Relative Pressure Accuracy	0.12 hPa
RMS Noise	0.2 Pa
Output Data Rate	25 Hz

STEREOLABS www.stereolabs.com

Figure 23 AF2. Camera Sensor Datasheet

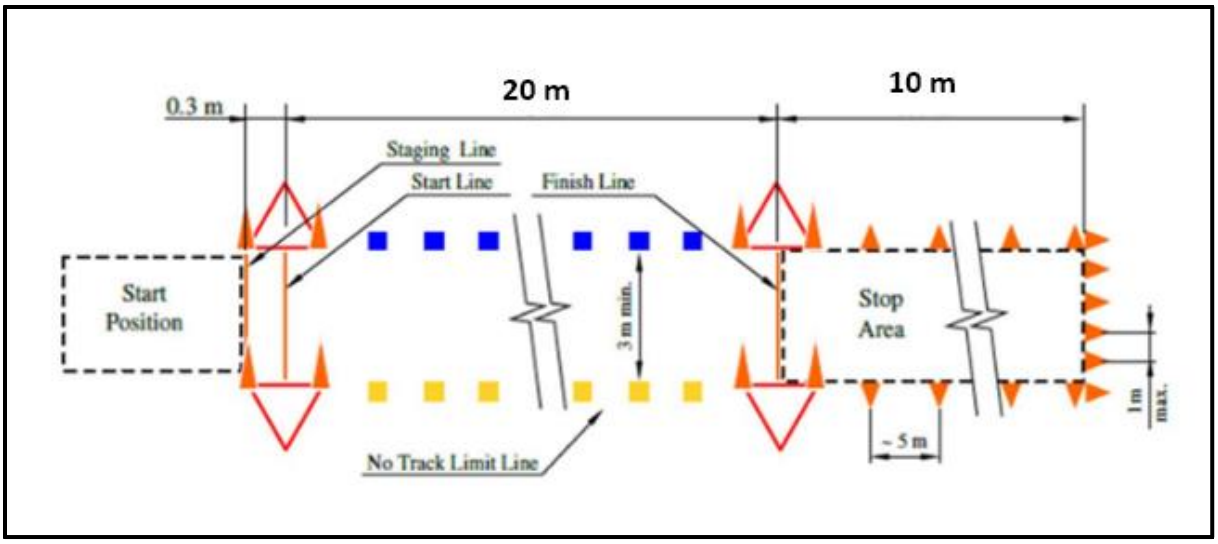


Figure 24 AF3. Acceleration Test Track Specifications

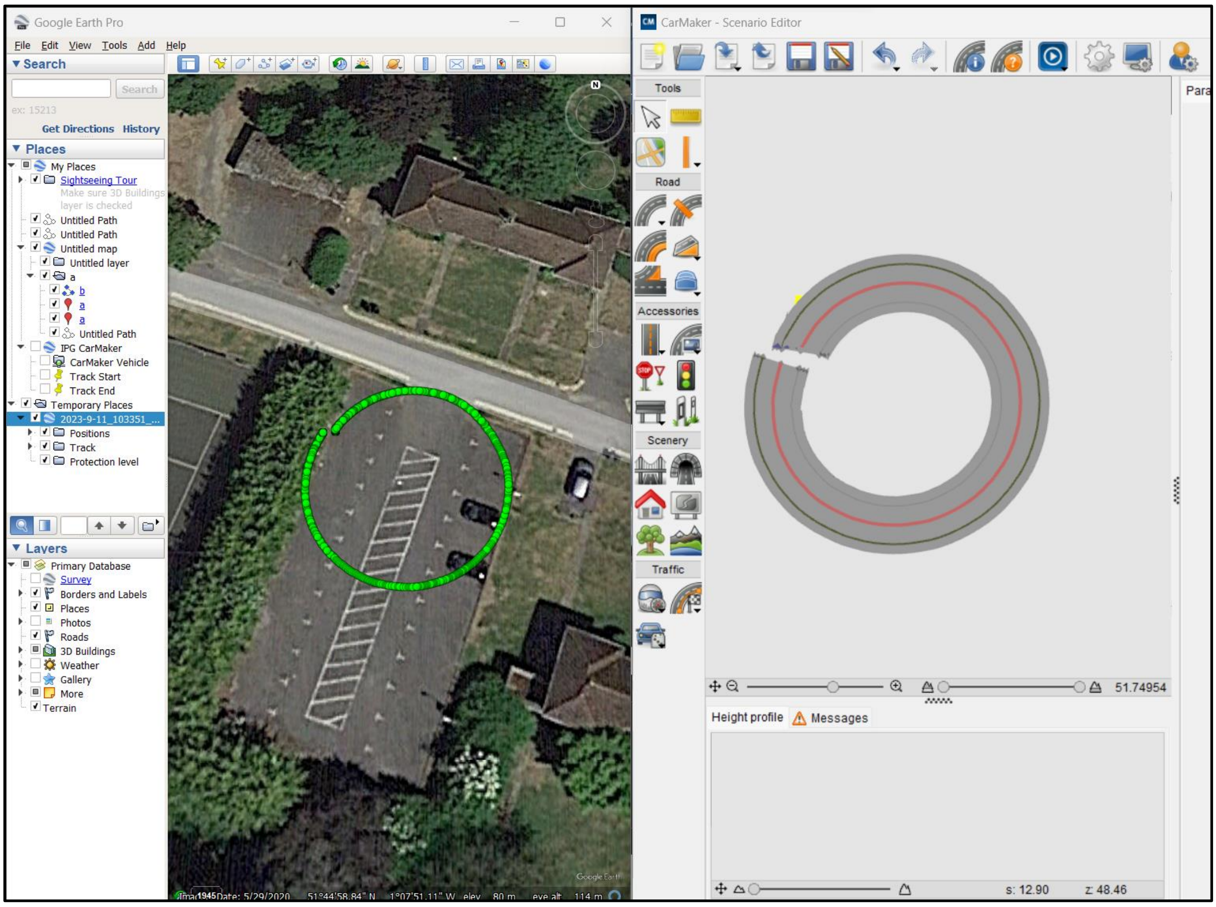


Figure 25 AF4. Physical Testing GPS co-ordinates used to develop test tracks in the Simulation Environment

ROS Bags

ROS bags are a command line tool that records data published on the system's topics (ROS, 2022). The ROS bag can contain various types of information, including sensor data, images, point clouds and robot state information. The recorded data can be replayed, visualised and analysed, allowing the developers to fine-tune the robotic systems and algorithms. To extract and save the data in the required format, a Python script written by an OBRA member was utilised.

A3. Results and Discussion

A3.1 Validation

	A	B	C
1	Wheel Static Measurment		
2	Degree	FL	FR
3	0	0	0
4	4	4	4
5	8	8	8
6	12	11	10
7	16	16	12
8	20	19	15
9	21	21	16
10	-4	-4	-3
11	-8	-7	-6.5
12	-12	-11	-10
13	-16	-16	-15
14	-20	-20	-18
15	-21	-21	-21

Figure 26 AF5. Static ADS-DV Steering wheel test response

			CLOCKWISE		
RADIUS	**Measured Angle**	**FL Wheel Real-world**	**FR Wheel Real-world**	**FL Wheel IPG**	**FR Wheel IPG**
(m)	(Deg)	(Deg)	(Deg)	(Deg)	(Deg)
0	0	0	0	0	0
4.40	19.00	21.00	17.00	21.15	19.49
5.95	14.50	15.75	13.25	16.39	15.36
9.45	8.75	8.84	8.66	9.89	9.49
			Anti-clockwise		
RADIUS	**Angle**	**FL Wheel Real-world**	**FR Wheel Real-world**	**FL Wheel IPG**	**FR Wheel IPG**
(m)	(Deg)	(Deg)	(Deg)	(Deg)	(Deg)
0	0	0	0	0	0
4.40	19.00	20.00	18.00	21.25	19.44
5.95	14.50	15.00	14.00	15.69	14.64
9.45	8.75	9.05	8.45	9.54	9.08

Table 10 AT1. ADS-DV Steering System Validation

Sensor Validation

The file was downloaded from **https://cddis.nasa.gov/Data_and_Derived_Products/GNSS/broadcast_ephemeris_data.html**.

a. GPS data of vehicle path in simulation

b. Scale down image of the Test track

Figure 27 A6. GPS Sensor Validation

Acknowledgements

As an author of this project, I would like to express my sincere gratitude and appreciation to all those who have contributed to the successful completion of this project. It is with immense pleasure and humbleness that I acknowledge the support, guidance, and encouragement received from various individuals and institutions.

First and foremost, I would like to express my heartfelt thanks to Dr. Andrew Bradley, my project supervisor and Faculty Advisor, whose expertise, constant guidance and valuable insights have been instrumental in shaping this project. His unwavering support and motivation have inspired the OBR Autonomous team and me to excel and explore new horizons.

I would also like to sincerely thank the team members of OBR-Autonomous, personally Aduen Benjumea and Sebastian Donnely, who were an indispensable part of this journey. The project was academically engaging and fun because of their commitment, teamwork, and companionship.

Finally, I thank my family and friends for their unwavering support, understanding, and encouragement throughout this endeavour. Their belief in me has been a constant source of strength, enabling me to overcome challenges and stay focused on achieving this objective.